THE UNIVERSITY
OF ADELAIDE
AUSTRALIA

SCHOOL OF BIOLOGICAL SCIENCES
DISCIPLINE OF BIOCHEMISTRY

The Design, Synthesis and Quantitative Analysis of a Bistable Mixed Feedback Loop Gene Network

Julian Michael Juers Pietsch

Thesis submitted in fulfilment of the requirements for the degree of

Doctor of Philosophy

February 2015

Abstract

Bistability, the capacity for switch-like memory, is a fundamental building block for robust behaviour in the noisy biochemical environment of a cell. Bistability has been observed experimentally in gene networks that exhibit overall positive feedback in some form; particular properties are endowed by variations on the basic network topology. The Mixed Feedback Loop (MFL) is a two-protein network that can be configured for positive feedback, and is notable since it has been observed to arise in nature more often than expected. The MFL includes an intervening protein-protein interaction to close a transcriptional feedback loop. This network architecture has been predicted to support bistable operation even without molecular cooperativity. To investigate the capabilities and features of the MFL, a synthetic bistable MFL was designed for construction in *Escherichia coli* (*E. coli*) using genetic components from bacteriophage 186. The design consists of the phage CI repressor protein inhibiting the production of its corresponding Tum antirepressor. This Tum−CI MFL prototype was first validated using a deterministic model expressly formulated for this instance of the MFL. It was then constructed in *E. coli* with dual LacZ and fluorescent reporters to permit multiple modes of measurement. Hysteresis assays — assays testing for history dependence or 'memory' of the system — were chosen as the measure of bistability, both since the bistable MFL naturally lends itself to such an assay, and since the assay simultaneously enables optimisation and setting of the switch. Measured by LacZ assay, the bistable MFL showed limited hysteresis. A detailed experimental characterisation of the network components and strains assisted in refining the data and setting bounds on model parameters. However, whilst this served to increase analytical accuracy, the deterministic model remained a poor fit of the data. When instead measuring activities in single cells by flow cytometry using the fluorescent reporter, two semi-stable sub-populations were discovered. Poor separation of the sub-populations necessitated the development of a system-specific mixture model for accurate identification of their characteristics, but the sub-population dynamics found much better agreement with the deterministic model. By building on this model with a hybrid stochastic/deterministic model, the limited hysteresis seen by LacZ assay can be explained by variation in switch robustness: the steady-state repressor concentration weights each cell's 'decision' for either of the two stable states. These results further an understanding of the core requirements for stable maintenance of epigenetic memory. The simplifications made by isolating the MFL according to the 'synthetic biology' approach allowed key features of this network motif to be determined. A deep knowledge of simple circuit structures like the MFL contributes fundamentally towards the way we understand proteins and how they fit into the complex networks that underpin the workings of life.

I certify that this work contains no material which has been accepted for the award of any other degree or diploma in my name in any university or other tertiary institution and, to the best of my knowledge and belief, contains no material previously published or written by another person, except where due reference has been made in the text. In addition, I certify that no part of this work will, in the future, be used in a submission in my name for any other degree or diploma in any university or other tertiary institution without the prior approval of the University of Adelaide and where applicable, any partner institution responsible for the joint award of this degree.

I give consent to this copy of my thesis when deposited in the University Library, being made available for loan and photocopying, subject to the provisions of the Copyright Act 1968.

The author acknowledges that copyright of published works contained within this thesis resides with the copyright holder(s) of those works.

I also give permission for the digital version of my thesis to be made available on the web, via the University's digital research repository, the Library Search and also through web search engines, unless permission has been granted by the University to restrict access for a period of time.

Julian Pietsch Date

Acknowledgements

This thesis would not have been possible without the generous assistance, support and patience of many others.

A number of colleagues were instrumental in guiding the development of the theoretical models and analytical methods employed in this thesis. The stochastic models started their development under the guidance of Kim Sneppen at the Niels Bohr Institute in Copenhagen. Sandeep Krishna, also from the Niels Bohr Institute, provided many thoughtful discussions on modelling techniques, which helped me to refine and simplify the deterministic models. Ray Correll (originally CSIRO) provided useful advice on advanced statistical practice and methods of inference, particularly for the analysis of the bacterial growth curves.

Many, many thanks must go to my supervisors Keith Shearwin and Ian Dodd, who were always excited about my work, but equally inspiring for the way they tempered my inventive imagination with sound wisdom. Thank you for tirelessly reading through many lengthy thesis drafts and for giving me the opportunity to work on such an enjoyable project.

Many past and present members of the Shearwin lab provided valuable experimental tips, reagents when I was short, and a satisfying source of good conversation. Rachel Schubert and Linda Shearwin provided a number of foundational materials and results for the work on the Tum protein. Adam Palmer and Michael Pocock started the MFL project, providing a solid cornerstone upon which much of this thesis has been built. Barry Egan, the previous head of the lab, provided encouragement and an infectious enthusiasm for phage 186. Alexandra Ahlgren-Berg, Erin Cutts, David Priest, Danna Li, Cui Lun and Andrew Hao made sure that I always had someone to chat with and participated in many hours of discussion at the lab-ins I presented. Thank you for being there for me through the highs and lows of the thesis. Many thanks must also go to Iain Murchland, who kept me up to date with the affairs of the outside world, entertained me with plenty of cheerful banter whilst at the bench, and was always willing to hear through my (often fanciful) thoughts and ideas.

In the final throes of the thesis, Pierre Dumuid helped me to steer back on course and establish realistic goals. Many thanks for standing with me in the face of a mammoth task.

My siblings Antony and Hannah Pietsch, and Taria and Adam Schulze have always given me something to smile about and look forwards to. Thank you for all the love and support you have shown me over the past few years, and for so many laughs and dinners.

My mum and dad, Helen and Mike Pietsch, have spoiled me with their generous love and care for many, many years. Thank you for being such amazing parents.

I owe so very much to my wife Alia Pietsch. You are my inspiration, my light. And your patience and love always carried me through the PhD's darkest moments. Thank you for believing in me.

Contents

List of Figures

List of Tables

List of Boxes

1

Introduction

The cell stands at the heart of the living world as one of natures most versatile building blocks. These packages of DNA, RNA and protein show a remarkably diverse range of behaviours and functions, from the efficient and rapid replication of the bacterial cell to the community of cells that make up the human body. Amazingly, this complexity is encoded in each cell's DNA using just a four-letter alphabet that prescribes the regulated production of RNAs and proteins, the chief workhorses of the cell. Understanding how these fundamental components can coordinate cellular behaviour is a key challenge facing the life sciences, with central relevance in understanding issues like how normal cells turn cancerous or how some pathogenic bacteria can alternate between dormant and aggressive disease-causing states.

To learn the principles upon which such complex systems are founded, it is often effective to begin by studying simple examples. The relative simplicity of bacteria have made them popular model organisms for the general study of cell behaviour. Though only small, single-celled organisms, bacteria are highly capable, being able to process their environment and make decisions. Lacking a network of neuronal cells, the 'brain' of a bacterium is instead the network of interactions between the many molecules inside the cell. The network of protein-DNA interactions, that is, the transcription regulation network, has traditionally been the best studied. However, interesting new patterns of network connectivity are emerging that involve a combination of protein-DNA and protein-protein interactions. One of the simplest examples, the MFL, involves a protein-DNA interaction that feeds back on itself via a protein-protein interaction.

Bacteria can 'remember' past events and use these to inform future decisions. The simplest example of this is digital memory — a bistable switch that can remember one of two alternative states. Such bistability can be achieved in a number of ways, but a common element is positive feedback in some form. There are many examples of positive feedback in natural networks, but these are typically integrated as part of a much larger network. To better understand memory in the cell, synthetic networks have been constructed in an attempt to discover the requirements for bistability by building it ourselves. Such synthetic systems have already proven to be quite informative, but they also present advances in the field of synthetic biology, which makes use of the rich functionality of biological systems

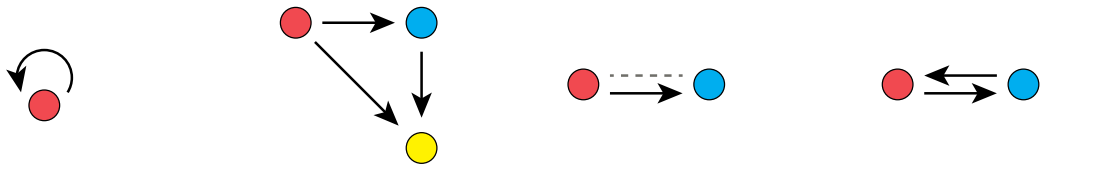
for the creation of new systems that can be applied in medicine and industry.

The MFL has the capacity to behave as a memory module, but has not yet been studied in isolation as a synthetic circuit. This thesis seeks to construct a synthetic bistable MFL in the bacterium *Escherichia coli* (*E. coli*) using parts obtained from bacteriophage.

1.1 Cellular networks drive cell behaviour

Just as the operation of an electronic circuit relies on the coordinated action of many interconnected components, the operation of a living cell relies on a vast and highly connected network of interactions between proteins, DNA, RNA and metabolites [Hartwell et al., 1999; Shen-Orr et al., 2002; Oltvai and Barabási, 2002; Joyce and Palsson, 2006]. This multilayered web drives the 'intelligent' behaviours exhibited by cells: from the way that a single bacterial cell can sense and move towards food (chemotaxis), to the commitment that human stem cells make towards a particular cell type (differentiation). The systems responsible for such behaviours can involve large numbers of interacting partners forming complex interaction networks. In spite of their complexity, however, cellular networks have evolved to favour high-order organisation [Alon, 2003; Barabasi and Oltvai, 2004; Alon, 2007; Mitra et al., 2013], and this has helped to direct research into the origins of cellular intelligence.

Electronic circuits are modular by design, and in a similar fashion cellular networks are punctuated by common 'network motifs': patterns of connectivity that are found more frequently than would be expected from randomised networks. Examples of some of the most common simple network motifs are depicted in Figure 1.1. Autoregulatory and feed forward motifs (Figures 1.1(a) and 1.1(b)) are common in the gene regulatory networks of both prokaryotes [Shen-Orr et al., 2002] and eukaryotes [Lee et al., 2002]. Motifs that cross traditional network boundaries are also prevalent, and their importance is being increasingly recognised [Ray et al., 2011; Mitra et al., 2013]. One of the simplest examples of this type of motif is the MFL depicted in Figure 1.1(c), which involves both gene and



(a) Autoregulation **(b)** Feed forward motif **(c)** Mixed feedback loop **(d)** Transcriptional feedback loop

Figure 1.1: Common network motifs in cells. Motifs are depicted using network diagrams, where circles (nodes) represent different genes in the network and connecting lines differ depending upon the type of interaction. For simplicity, each node represents both a gene and its protein product. Transcriptional regulation of one gene by the protein product of another is depicted using an arrow. Dashed lines represent a protein-protein interaction between gene products.

protein interactions. The MFL is known to be a significant motif in the yeast *Saccharomyces cerevisiae* (*S. cerevisiae*) and is also found in many other cell types ranging from bacterial to mammalian [Yeger-Lotem et al., 2004]. In contrast, the transcriptional feedback loop in Figure 1.1(d) is not well-represented in the typically fast-response networks of bacteria or yeast, but is a notable motif in the long-term developmental pathways of higher organisms [Alon, 2007]. In their respective contexts, the over-abundance of each of these motifs implies that they may capture some core processing functionality, and much attention has been given both to their discovery in other species and to their characterisation, as reviewed by Alon [2007].

Such studies are of broad impact: network architecture is thought to be better conserved between species than the constituent components are [Oltvai and Barabási, 2002], with some structural similarities even being shared with unrelated networks like food webs, social networks and the internet [Milo et al., 2002]. As such, many of the principles learned by studying small networks are transferable, making it possible to study larger networks from the bottom up [Guido et al., 2006]. Indeed, much of our current understanding of cellular networks builds upon concepts developed last century using the relatively simple networks of the model organism bacteriophage λ (phage λ) and its host, the bacterium *E. coli*. An excellent overview of this body of research for phage λ can be found in the book by Ptashne [2004], but a brief summary follows. λ is a temperate phage — a virus that infects bacteria and that, upon infection, decides between two alternative lifecycles. In one, the lytic pathway, it uses host machinery for aggressive replication and packaging of its DNA into phage particles, which are eventually released by bursting (lysing) the host. In the other, the lysogenic pathway, the phage DNA reproduces passively with replication of the host by integrating (inserting) its genome into the host chromosome and keeping the genes responsible for lytic growth silent. The decision between these two pathways is controlled by a small network of transcription factors. The core motif of this network is an example of the transcriptional feedback loop (Figure 1.1(d)) found in developmental pathways, and its study has been instrumental in building an understanding of how network architecture can translate into a committed decision between two states. Furthermore, in spite of its relative simplicity, the transcriptional network of phage λ continues to shed light on cell regulation and memory [Dodd et al., 2004; Zeng et al., 2010; Zong et al., 2010; Cui et al., 2013].

Rigorous studies of the roles and functions of network motifs stand as pivotal steps towards understanding cellular behaviour. Amongst the ways for studying network motifs, the methods of synthetic biology stand as some of the most promising, and are worth developing in their own right.

1.2 Synthetic biology as a tool for studying network motifs

Synthetic biology is an emerging field whose aim is the creation of new biological functions and organisms. Living systems have evolved a plethora of elegant solutions to the challenges posed by nature. By building upon this diverse and highly capable toolkit to engineer solutions to humanity's own challenges, synthetic biology holds the potential for landmark applications in both industry and medicine. Though still a young field, promising applications have already come to fruition. The engineering and optimisation of a metabolic pathway for biosynthesis of artemisinic acid in *S. cerevisiae* [Ro et al., 2006] has paved the way for large-scale production of the antimalarial compound artemisinin [Paddon et al., 2013]. Bacteriophage have been engineered to break down bacterial biofilms that would normally act as a protective barrier for pathogenic strains [Lu and Collins, 2007]; others have been engineered to enhance the efficiency of antibiotics at eliminating infections [Lu and Collins, 2009]. Bacteria have been synthesised that can specifically target and deliver cytotoxic payloads to cancerous cells [Anderson et al., 2006; Huh et al., 2013], or that can stimulate the immune system to eliminate human lymphomas [Massa et al., 2013]. In response to the threat of global warming, much effort has also been spent in engineering microbes for the sustainable production of biofuels as reviewed by Kung et al. [2012]. Many other applications of synthetic biology are in development and covered in more detail in the reviews by Khalil and Collins [2010] and Ruder et al. [2011]. Synthetic biology holds many exciting applications, but important accompanying functions of these endeavours are the lessons learned about nature's design rules.

Much like synthetic chemistry helped to establish the principles of chemical reactivity and molecular structure, the process of building novel gene circuits tests and can help to refine our understanding of network architecture [Yeh and Lim, 2007]. By rearranging existing components to test their presumed purpose, or by attempting to recreate observed behaviours using unnatural components, synthetic biology can help to identify core biological functions that are easily missed by observation and analysis alone [Benner and Sismour, 2005]. The simple motifs found in gene networks are prime candidates for reconstruction as synthetic gene networks, and much has already been learned by this approach [Sprinzak and Elowitz, 2005; Mukherji and van Oudenaarden, 2009]. The development of key design standards has driven much of synthetic biology over the last decade [Way et al., 2014]. These include the identification of modularity in biology [Agapakis and Silver, 2009], the need for predictive mathematical models [Hasty et al., 2002], and the need for a standardised library of decoupled parts [Endy, 2005]. These concepts will be introduced in more detail in the coming sections.

1.2.1 Defining modules for rational circuit design

In rationally designing new systems, it is important to recognise parts that are modular: parts whose functions in a native context could be transferred intact into engineered environments. Modularity is present to varying degrees at almost all levels of operation of the cell, from the triple base pair codons that are translated into amino acids, to protein domains, genes and many of their control elements [Benner and Sismour, 2005]. Entire sub-networks can even be classified into modules by associating network nodes that share a common purpose [Mitra et al., 2013].

Modules are the synthetic biologist's building blocks, and in gene network engineering, some of the most important are transcription factor genes and the promoters they regulate. Well-characterised transcription factors, like the CI repressor protein from phage λ, the TetR repressor protein that regulates resistance to the antibiotic tetracycline, and the LacI repressor protein that regulates lactose metabolism in *E. coli*, have seen extensive use in synthetic gene networks. The earliest synthetic gene networks were built from novel rearrangements of just these three transcription factors and their respective promoters to create small networks with a diverse range of behaviours and architectures, including a switch [Gardner et al., 2000], an oscillator [Elowitz and Leibler, 2000], logic gates [Guet et al., 2002] and autoregulatory motifs [Becskei and Serrano, 2000; Rosenfeld et al., 2002]. The reusability of these parts demonstrates their modularity, and indeed these regulatory networks can be interfaced with other components to predictably alter sensory inputs and phenotypic outputs [Kobayashi et al., 2004]. Furthermore, these parts were used to replace functionally analagous ones in the phage λ lysis-lysogeny switch, and the synthetic network shown to reproduce the salient behaviour of the natural system [Atsumi and Little, 2006].

The success of these early synthetic networks showed that transcription factor genes and their promoters could be treated as modular and rewired by DNA recombination to produce desired behaviours. These were, however, exploratory studies that produced qualitative rather than quantitative agreement with the design specifications. It has since become apparent that the point-blank abstraction of these parts as autonomous modules with singular functions is, in many cases, too crude a simplification for the robust development of new network architectures [Andrianantoandro et al., 2006; Nandagopal and Elowitz, 2011]. This has prompted a more careful and detailed treatment of the modules used in synthetic biology.

In their native contexts, most proteins and regulatory elements are tightly integrated within the broader cell network, displaying multiple, often overlapping functions. Even the CI repressor of phage λ, a frequent component of synthetic networks, has multiple functions: depending on the genetic context, it can act either as a transcriptional repressor or activator of promoter elements, and it is also linked to the host SOS response network, in which it gets cleaved and thereby inactivated upon SOS signalling [Ptashne, 2004]. Avoid-

ing such coupling in the design of new networks has until more recently been a common theme in synthetic biology [Nandagopal and Elowitz, 2011]. Though it might be technically possible to create completely specified environments by rewriting the entire genome of an organism [Gibson et al., 2008, 2010], or by constructing minimal synthetic systems *in vitro* [Kim et al., 2006], the engineering and analysis of small synthetic gene circuits is most easily achieved using an existing organism as the 'host'. This means that global cell regulatory factors are an important consideration in circuit design. For example, the introduction of new genes can result in changes to growth rate, with global consequences that can affect quantitative and qualitative behaviours of the gene circuit [Klumpp et al., 2009]. Designs may even need to account for coupling between modules: wiring the output of one module to the input of the next can have unintended effects on the dynamics of the upstream module — a phenomenon known as retroactivity [Del Vecchio et al., 2008; Jayanthi et al., 2013].

Instead of presenting a hindrance to circuit design, coupling between modules and coupling with global host factors can present opportunities for novel regulatory mechanisms. Synthetic circuits can be constructed using factors known to modulate growth rate, and this modulation then exploited for the production of a memory network [Tan et al., 2009]. The generation of oxidative radicals by the fluorescent reporter proteins normally used to track cell state can be used to engineer long-distance quorum sensing behaviour [Prindle et al., 2012]. Different proteins that are actively degraded by the same host protease can be coupled by saturation of that enzyme, and this coupling used to create an oscillator [Prindle et al., 2014].

Nonetheless, many instances of coupling are unexpected in the early stages of network design, making the process of synthetic biology naturally an iterative one [Sprinzak and Elowitz, 2005]. Yet with the construction of each new synthetic network, the arsenal of mitigation strategies in the synthetic biologist's toolkit keeps expanding. These strategies are reviewed in more detail by Brophy and Voigt [2014], but include an increasing variety of methods for tuning module behaviour, such as the adjustment of promoter strengths, ribosome binding site (RBS) strengths, rates of degradation or gene dosage (e.g., by adjusting plasmid copy number). Even the effects of retroactivity can be overcome through the judicious application of time scale separation [Mishra et al., 2014]. Ideally, these improvements could all be made in the design process itself by refining the mathematical models that are used to predict network behaviour.

1.2.2 Predictive models of cellular networks

Quantitative modelling provides a rigorous framework for developing and testing ideas about the behaviour of gene networks. Indeed, a quantitative appreciation of a network's components is essential for accurately determining its dynamical behaviour [Ronen et al., 2002]. Even the seemingly simple coupling of regulators with effectors in inducible and

repressible circuits can see *qualitative* changes arising out of *quantitative* ones [Hlavacek and Savageau, 1996; Wall et al., 2003]. As such, mathematical models are particularly influential in synthetic biology, where they can benefit both the design and evaluation of synthetic gene networks [Hasty et al., 2001, 2002]. Even in the very first synthetic gene networks, mathematical models were instrumental in the design process [Elowitz and Leibler, 2000; Gardner et al., 2000]. Though the models in these early studies were not quantitatively accurate, they could be used to identify possible classes of dynamic behaviour and thereby inform the optimisation of experimental parameters. Since then, improvements in the experimental quantitation of network components and in the models used to describe them have resulted in circuit designs where the calibrated model can accurately predict the quantitative output of the engineered network [Isaacs et al., 2003; Guido et al., 2006; Rosenfeld et al., 2007; Ellis et al., 2009b].

Gene networks can be characterised using many different types of modelling depending on the desired level of detail [Karlebach and Shamir, 2008]. Of these, deterministic rate equation models and a variety of stochastic models have seen regular use alongside synthetic networks [Kaern et al., 2003]. Deterministic models track network components like messenger RNA (mRNA) and proteins as continuously varying concentrations that can be defined in terms of an Ordinary Differential Equation (ODE). Deterministic models are popular in formulating the initial network design, since their analytical tractability helps to simplify the process of exploring suitable parameter regimes [Gardner et al., 2000; Rosenfeld et al., 2002; Atkinson et al., 2003; Tigges et al., 2009; Palani and Sarkar, 2011]. Such models are kept deliberately simple to facilitate fast prototyping of designs, whilst simulaneously avoiding the typically poorly-defined details of many biochemical reactions. This loss of accuracy does not necessarily reduce their utility: simplified models are quite likely to be valuable in summing up the behaviours of large networks from the behaviours of their motifs [Sneppen et al., 2010].

In contrast, stochastic models provide a more accurate account of network dynamics at the cost of simplicity. Cellular environments are noisy: at such small scales, the intrinsically random timing of biochemical reactions can become significant, prompting the treatment of transcription and translation as stochastic reactions [Ozbudak et al., 2002; Elowitz et al., 2002]. Stochastic models are frequently applied after characterisation of a synthetic circuit to explain its noisy behaviour, or to investigate network mechanics in more detail [Elowitz and Leibler, 2000; Becskei and Serrano, 2000; Isaacs et al., 2003; Hooshangi et al., 2005; Stricker et al., 2008]. Methods for stochastic simulation continue to be improved, and can in some cases be applied to accurately predict the behaviour of synthetic networks [Guido et al., 2006]. Furthermore, our increased understanding of different stochastic mechanisms have led to advances in network analysis: distributions of noise can provide additional information about regulatory activity, and by developing an analytical framework through the use of simple synthetic circuits, the behaviour of natural

gene networks can be better understood [Dunlop et al., 2008].

Irrespective of the choice of model, careful characterisation of the components is essential to a model's predictive success. This highlights another fundamental goal of synthetic biology: the collation of a library of well-characterised parts that can be rationally pieced together with the assistance of both deterministic and stochastic models.

1.2.3 Bacteriophage 186: a source of new components for synthetic biology

The modular interpretation of the cell paved the way for many landmark synthetic gene networks, whose success depended on a core collection of well-studied parts [Voigt, 2006]. However, the limited number of such parts and the absence of systematic methods for their characterisation soon made it clear that standardised component libraries would be required for continued growth in the field [Endy, 2005]. These observations prompted the development of rigorous standards for the specification of parts [Canton et al., 2008], and the development of a comprehensive and accessible repository of interchangeable parts [Shetty et al., 2008]. Whilst these represented major steps forwards in bioengineering capacity, the next generation of synthetic networks are set to demand increasingly large collections of orthogonal components [Lu et al., 2009]. Significant progress has already been made in generating large promoter and RBS libraries with a broad range of strengths [Ellis et al., 2009b; Mutalik et al., 2013]. Such libraries can be used for predictable tuning of expression levels when optimising synthetic networks, but do not contribute to the diversity of network connectivity. Libraries of network components are most easily obtained using the wealth of parts present already in nature. For example, it became possible to synthesise a large network containing multiple logic gates by finding homologues of the required components in different strains of bacteria, making use of directed evolution to optimise these for dynamic range and orthogonality (i.e., reduced coupling) [Moon et al., 2012].

Viruses, including bacteriophage, are one of the planet's most abundant organisms, and stand as an especially diverse source of genetic material [Rosario and Breitbart, 2011]. As was previously observed in the case of phage λ, the relative simplicity of bacteriophage makes them useful model organisms, but it also makes them convenient sources of parts for synthetic biology. Temperate phage, in particular, contain components that are valuable in network design, since they require a memory circuit to maintain the lysogenic state of passive replication with the host. A temperate phage that has been studied in detail over the last three decades is the bacteriophage 186 (phage 186) [Woods and Egan, 1974; Kalionis et al., 1986; Dodd et al., 1990; Neufing et al., 2001; Dodd et al., 2007b]. This phage displays many functional similarities with phage λ, but is remarkable for its distinct regulatory mechanisms [Trusina et al., 2005]. This has made phage 186 a useful counterpoint to phage λ and makes it an exciting new source of synthetic parts, especially since many of its components have already been well-characterised.

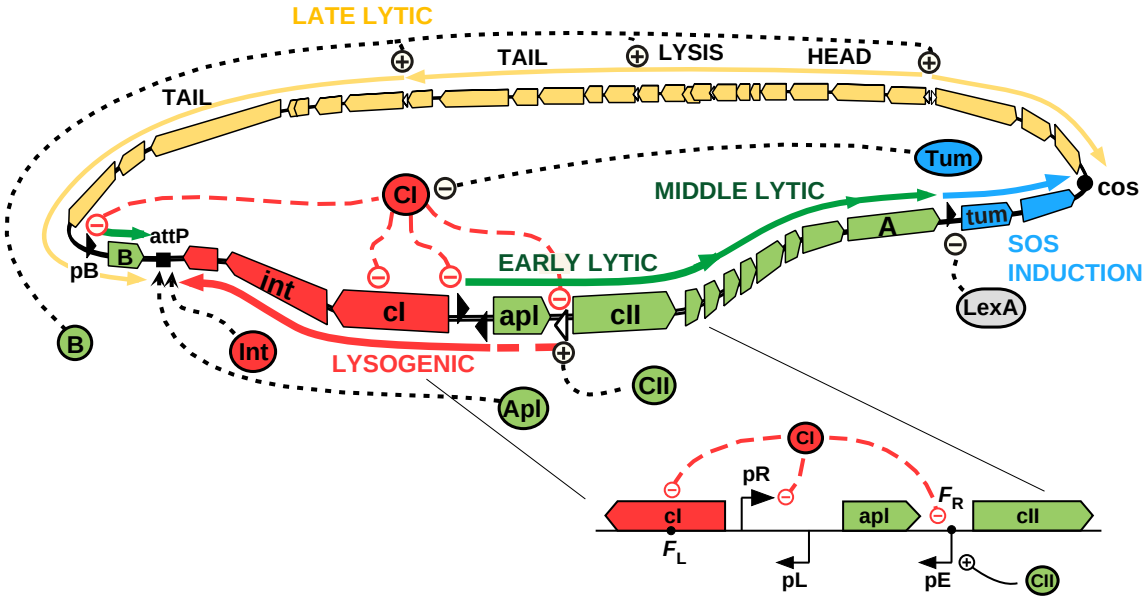


Figure 1.2: A diagrammatic representation of the bacteriophage 186 genome. The entire genome consists of only 30.6 kilobases of DNA, but encodes for all the functionality of the phage. The inset provides more detail for the main control region. Points of CI binding are indicated by the red heavy dashed lines; the binding sites flanking the *pR* promoter (F_L and F_R) are indicated by a black dot in the inset. See the text for descriptions of the most relevant genes and proteins. (Adapted from a figure by Ian Dodd).

Two major distinguishing features of phage 186 are its switch control region and its mechanism of induction in response to host SOS signals. To illustrate these differences by reference to the gene network of phage 186, a schematic of the entire phage genome is given in Figure 1.2, with additional details for the switch control region shown in the inset. Like phage λ, a repressor protein called CI[1] maintains the lysogenic state by keeping the lytic promoter *pR* silent [Lamont et al., 1993; Dodd and Egan, 2002]. However, unlike phage λ, the promoters that drive lysogenic and lytic development (*pL* and *pR* respectively) are located in a face-to-face arrangement. This means that repression of the lytic state is further enhanced by transcriptional interference — the repression of a promoter by incident RNA polymerases [Callen et al., 2004; Dodd et al., 2007b].

The mechanism of phage induction in response to SOS signalling is another point of contrast. The SOS network of *E. coli* gets activated in response to DNA damage [Little and Mount, 1982], and under such conditions, lysogens of phage λ and phage 186 are induced into lytic development by the inactivation of their repressor proteins. In the case of phage λ, inactivation of the repressor occurs by stimulated autocatalytic cleavage of λ CI by the host RecA protein [Little, 1984]. In the case of phage 186, response to the host SOS system is mediated by the p_{95} promoter that is normally held repressed by the host LexA protein [Brumby et al., 1996]. Upon SOS signalling, LexA is inactivated and expression of the *tum* gene produces the antirepressor Tum. Tum reversibly inactivates CI by preventing

[1]Note that unless otherwise indicated, references to *c*I or CI in this thesis will be used to refer to that gene or protein respectively in phage 186, rather than to the identically named repressor protein in phage λ.

its binding to the *pR* promoter and thereby allowing lytic transcription [Shearwin et al., 1998]. The precise mechanism of CI inactivation by Tum is unknown, but it is known that (1) the Tum protein acts on CI without the aid of any other factors, (2) the effect is Tum concentration-dependent, and (3) the action of Tum on CI is reversible [Shearwin et al., 1998]. So for the purposes of network design, CI can be thought of as a repressor protein for the *pR* promoter, and Tum as an antirepressor that interacts with CI as if by a protein-protein interaction. This makes these two components ideal candidates for studying the intersection of transcriptional and protein-protein interaction networks.

1.3 The bistable MFL is an excellent candidate synthetic network

Cellular networks that can exhibit bistability — that is, an ability to stably maintain either of two alternative states — have attracted much attention, since such a property can give a cell the capacity to exhibit committed all-or-none responses. Natural examples of bistable networks include the lysis-lysogeny decision of phage λ [Oppenheim et al., 2005], the regulation of lactose metabolism in *E. coli* [Ozbudak et al., 2004], the persistence of antibiotic-resistant bacteria in the treatment of infection [Balaban et al., 2004], the transient competence of *Bacillus subtilis* (*B. subtilis*) [Süel et al., 2006], the differentiation of the photoreceptor cells responsible for colour vision in *Drosophila* [Mikeladze-Dvali et al., 2005], and the epigenetic inheritance of nucleosome modifications [Dodd et al., 2007a] and DNA methylation [Lim and van Oudenaarden, 2007]. This widespread incorporation of bistability in natural networks underscores their importance in building complex network behaviours, and real-world medical and industrial applications of bistable synthetic networks are anticipated [Burrill and Silver, 2010]. Precedent for potential medical applications was recently set with the synthesis of a diagnostic bacterial strain: by making use of the bistable phage λ switch, the synthetic strain could remember diagnostic signals detected while passing through the mouse gut [Kotula et al., 2014].

Synthetic bistable networks have also played significant roles in advancing our understanding of bistable behaviour and the requirements for bistability. Bistability requires both overall positive feedback, but also some form of nonlinearity [Ferrell, 2002]. Examples of gene regulatory networks with overall positive feedback are illustrated in Figure 1.3. The first synthetic bistable network was built using the toggle switch design (Figure 1.3(b)) [Gardner et al., 2000]. The toggle switch is very similar to the bistable motif of phage λ, and derives nonlinearity from the cooperative binding of the repressor proteins to their promoters. It is a mutually exclusive switch: in one state only A is expressed, since A represses the expression of B; in the other state only B is expressed. Toggle switches have since been synthesised in mammalian cells [Kramer et al., 2004; Kramer and Fussenegger, 2005] and also in a minimal *in vitro* system [Kim et al., 2006]. An enhanced mammalian toggle switch was also synthesised, in which the stability of each state was enhanced through

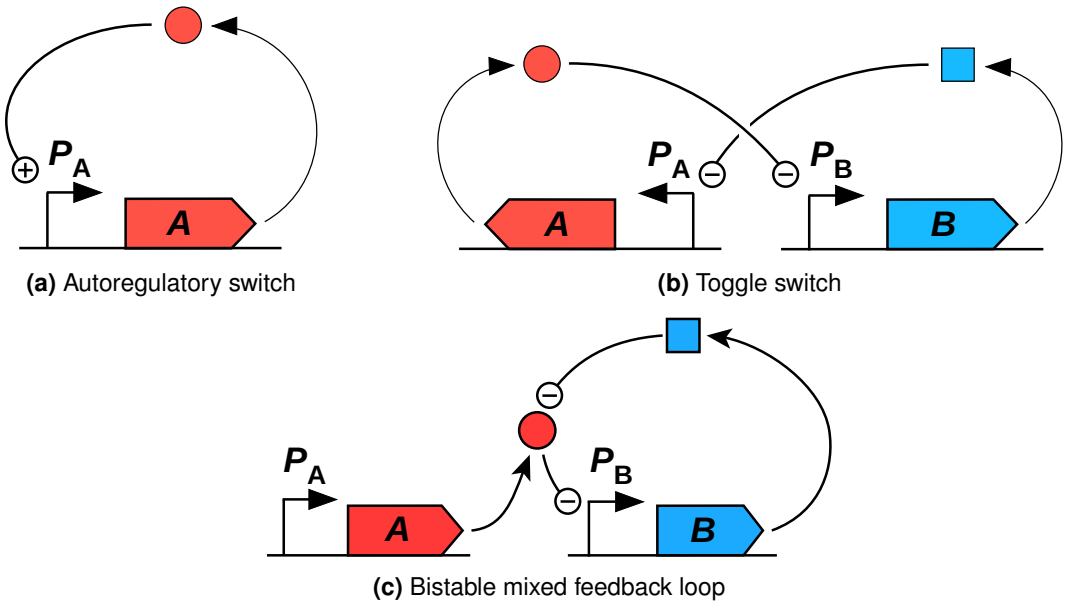


Figure 1.3: Gene regulatory networks that can exhibit bistability. Promoters are depicted as bent arrows, genes as directed boxes and proteins as circles or squares. **(a)** Positive autoregulation where protein A activates its own transcription. The switch can exist in high A or low A states. **(b)** The genetic toggle switch where protein A represses transcription of protein B, and protein B similarly represses transcription of protein A. The switch can exist in high A/low B or low A/high B states. **(c)** The bistable mixed feedback loop where protein A represses transcription of protein B, whilst protein B interacts with protein A to prevent its repression of the P_B promoter. The switch can exist in high B or low B states.

an additional feedback mechanism involving mRNA silencing by siRNAs (short interfering RNA molecules) [Greber et al., 2008].

Shortly following construction of the original toggle switch, a bistable circuit based on the positive autoregulation design (Figure 1.3(a)) was constructed in *S. cerevisiae* [Becskei et al., 2001]. Such circuits also derive nonlinearity from cooperative binding. An autoregulatory switch was also synthesised in *E. coli*, which became a robust oscillator with the addition of a linked negative feedback loop [Atkinson et al., 2003]. A particularly stable switch was engineered by instead linking positive autoregulation with the bistable motif of the *lac* operon [Chang et al., 2010]. In fact, more generally, the linking of fast and slow positive feedback loops is believed to be an important means for increasing robustness in bistable networks [Brandman et al., 2005].

In *E. coli*, the toggle switch is not a common motif [Shen-Orr et al., 2002], and positive autoregulation is much less common than negative autoregulation [Thieffry et al., 1998]. Rather, composite feedback loops involving a transcriptional and post-transcriptional link are thought to be more important [Shen-Orr et al., 2002], as they are in *S. cerevisiae* [Yeger-Lotem and Margalit, 2003; Yeger-Lotem et al., 2004]. One of the most common composite motifs is a feedback loop consisting of a protein-protein interaction and a protein-DNA interaction [Yeger-Lotem et al., 2004; Alon, 2007]. This MFL motif, introduced earlier in Section 1.1, has been shown in theory to support either bistability or oscillations depending

on the overall sign of the feedback [François and Hakim, 2005]. With overall negative feedback, the MFL can behave as an oscillator. In human cells, the tumour supressor p53 and its negative regulator Mdm2 form such a loop: p53 activates transcription of Mdm2 (positive interaction), whilst the Mdm2-p53 protein-protein interaction targets p53 for degradation (negative interaction) [Lahav et al., 2004]. The yeast galactose operon contains a MFL with overall negative feedback comprising the Gal4p activator and the Gal80p protein, but two additional positive feedback loops involving the sequestration of Gal80p by Gal3p and Gal1p make this network a bistable one [Smidtas et al., 2006; Venturelli et al., 2012].

With overall positive feedback the MFL can behave as a bistable switch [François and Hakim, 2005]. An example of the MFL motif in this regime is illustrated in Figure 1.1(c). In *B. subtilis*, the decision between chained growth in biofilms and motility is governed by such a bistable MFL. The SinR repressor controls production of SlrR, which in turn forms a complex with SinR that prevents it from repressing SlrR production [Chai et al., 2010a,b]. Another bistable MFL is found in the genetic switch of temperate bacteriophage TP901, where the CI repressor of that phage controls the production of MOR, an antirepressor that inhibits repression by CI [Nakanishi et al., 2009; Alsing et al., 2011]. In *S. cerevisiae*, the Swi4 and Swi6 proteins involved in cell cycle control also form an MFL with overall positive feedback, though this example is embedded within a much larger network [Baetz and Andrews, 1999].

Bistability in the MFL is possible without cooperativity in protein binding, since protein sequestration itself generates an ultrasensitive response [Buchler and Louis, 2008; Buchler and Cross, 2009]. Stochastic modelling of a theoretical MFL with overall positive feedback shows that both intrinsic and extrinsic sources of noise could be capable of causing switching between the two stable states, and, for certain parameter regimes, a noisy MFL may even support oscillations [Li and Li, 2008]. Whilst there are natural examples of the MFL embedded in larger networks and theoretical studies of the bistable MFL, a systematic study of a synthetic bistable MFL has not, to the author's knowledge, yet been performed. As such, the primary aim of this thesis is the construction and detailed characterisation of a synthetic bistable MFL using the repressor and antirepressor proteins of phage 186.

1.4 Thesis overview

A MFL with overall positive feedback could be made by placing the expression of the Tum antirepressor under the control of the *pR* promoter and supplying CI from another promoter. A proof-of-principle deterministic model of this Tum−CI MFL is developed in Chapter 2. By considering the deterministic model at steady-state, the capacity for bistability in the Tum−CI MFL is demonstrated for a range of different model parameters. The dynamical behaviour of the model is then used to propose an experimental design that

could demonstrate bistability by the traversal of a hysteresis loop. The cloning of various alternative Tum−CI MFL strains in *E. coli* is then described in Chapter 3. Traversal of the hysteresis loop is enabled by placing the expression of CI under the control of an inducible promoter. A dual reporting system is also devised so that the state of the switch — the activity of the *pR* promoter — can be tracked either by fluorescence or by LacZ assay. Preliminary experiments obtained by LacZ assay prompt better characterisation of the switch components, including the nonlinearity present in the induction system, and the production and degradation rates of Tum and CI.

Hysteresis of the experimental Tum−CI MFL is demonstrated in Chapter 4. The LacZ assay enables efficient screening of the different strain designs, but also reveals that the original hysteresis protocol does not allow enough time for equilibration of the network. By using a protocol that produces a longer equilibration time, hysteresis is confirmed in some strains but not in others. However, even in the successful strains, the observed hysteresis is weak. Neither the deterministic model nor the measured component properties could be used to explain this weak hysteresis. Instead, an explanation in terms of noise-induced instability is proposed. Measurements by LacZ assay give only the average *pR* activity of a large population of cells, so LacZ assay results could mask a more complicated underlying distribution of *pR* activities in individual cells. Thus in Chapter 5, measurements by flow cytometry are used to measure the *pR* activities of single cells using the fluorescent reporter. Evidence for a mixture of two cell populations in the middle of the hysteresis loop prompts the development of a mixture modelling method that can resolve these sub-populations. The mean activities of these sub-populations mark a much better match to the hysteresis curves predicted by the deterministic model. To explain the split into two sub-populations, however, a stochastic model is required. A basic hybrid deterministic/stochastic model is derived in Chapter 6 that can reproduce the split-population phenotype based on just the intrinsic noise in CI and Tum production. However, in order to match the level of noisy switching observed experimentally, larger than expected fluctuations in Tum expression are required, suggesting an explanation of switch instability based on plasmid copy-number fluctuations.

In Chapter 7, the synthetic Tum−CI MFL is compared with other natural and synthetic examples of bistability and future work on the Tum−CI MFL is proposed. Detailed experimental methods for the preparation and analysis of the Tum−CI MFL can be found in Chapter 8. The thesis concludes with a draft paper describing further characterisation of the Tum protein by mutation and structure prediction to better consolidate Tum as a useful part for future synthetic circuits.

2

Directing design of a bistable genetic circuit by mathematical modelling

The bistable Mixed Feedback Loop (MFL) as envisioned by François and Hakim [2005] is a two-component network consisting of a positive feedback loop in which a protein gene product indirectly regulates its own transcription via a protein-protein interaction. This network is schematically represented in Figure 2.1 with an antirepressor factor B that stimulates its own production from the promoter P_B by relieving repression of that promoter by the repressor A. François and Hakim developed a generalised dynamical model of the MFL, showing that with this overall positive feedback, the MFL operates under a bistable regime[1]. Such an antirepressor-repressor interaction is found between the Tum antirepressor and CI repressor of phage 186 [Shearwin et al., 1998]. In the phage, expression of Tum is activated by the host SOS response; Tum then reversibly acts against the CI repressor to relieve repression of phage lytic promoters such as *pR*. In principle, a bistable MFL could be constructed using these phage components by placing Tum production under control of the CI-repressible promoter *pR*. In this chapter, further design considerations regarding the practical implementation of a synthetic Tum−CI MFL network are explored. To this end, initial qualitative descriptions of this circuit and its behaviours are developed into quantitative models that consolidate and extend our understanding of the network.

[1] François and Hakim [2005] contrast the MFLs bistable regime with an oscillatory regime with overall negative feedback in which the the protein B relieves *activation* of its own promoter.

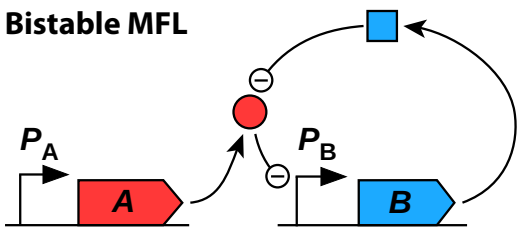


Figure 2.1: The bistable Mixed Feedback Loop (MFL) is a genetic network motif in which positive autoregulation of gene *B* proceeds via an intervening protein-protein interaction with *A*. The bistability of this genetic circuit depends on overall positive feedback, generated here by a double negative feedback loop where the antirepressor B relieves repression of its own promoter P_B by the constitutive repressor A.

2.1 Origin of bistability in the Tum−CI MFL

By definition, bistable systems are those for which there are two distinct stable states. A light switch shows bistable behaviour since it 'remembers' whether it is on or off. The switch can be flipped between the two states by outside interference (a finger), but once it enters either state, it stays there indefinitely. Note how this contrasts with a doorbell — a monostable system — which can be induced to an on state (by pressing the button), but always returns to its stable off state when the inducing signal is removed. In biochemical networks, these states are protein concentrations or promoter activities. In the bistable MFL, the 'on' state has a high concentration of protein B and high expression from P_B, whilst the 'off' state has a low concentration of protein B and low expression from P_B.

These two states are illustrated in Figure 2.2 for a MFL constructed from Tum and CI. In the 'on' state, production of the antirepressor Tum from the *pR* promoter is high, so cellular concentrations of Tum are high. This shifts the Tum−CI equilibrium towards formation of an inactive complex. The available CI is sequestered (i.e., locked away) and cannot act at the *pR* promoter so the 'on' state is maintained. In the 'off' state, transcription from *pR* is low, so cellular concentrations of Tum are also low. This shifts the Tum−CI equilibrium away from formation of the complex, and the available CI is then free to act at the *pR* promoter and maintain the 'off' state. Notice that in both 'on' and 'off' states, the level of CI is the same. This highlights the important point that in bistable networks, identical external states (fixed CI expression levels) can give rise to two alternative internal states (high or low Tum concentrations). Put another way, the equilibrium state of the bistable MFL cannot be predicted without first knowing its history.

Simultaneously achieving the capacity for two stable states requires balanced production rates. The 'on' state can only be stably maintained if high enough rates of Tum production can be reached to overcome repression of *pR* by the fraction of unbound CI. Conversely, the 'off' state can only be stably maintained if low enough rates of Tum production can be reached to prevent sequestration of the fraction of free CI.

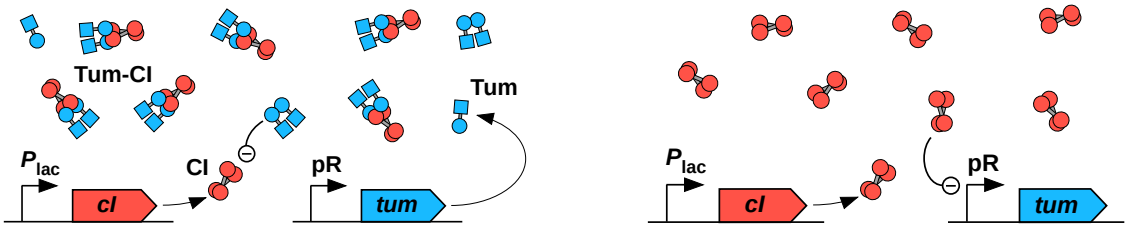


Figure 2.2: The two stable states of a bistable Tum−CI Mixed Feedback Loop. CI repressor produced from P_{lac} would normally repress the promoter *pR* that drives production of the antirepressor Tum. However, in the 'on' state Tum relieves this repression by sequestering CI and sustaining its own production. In the 'off' state, the same level of CI produces an alternative stable state where there is not enough Tum present to begin with to prevent repression of the *pR* promoter.

A source of feedback is necessary but not sufficient to guarantee bistability. A further, less obvious requirement, is ultrasensitivity [Ferrell, 2002], that is, feedback that proceeds via a nonlinear response curve that is sigmoidal or S-shaped. The simplest type of ultrasensitivity arises as a result of the cooperative binding of oligomers to produce a Hill curve response [Smolen et al., 2000; Qian, 2012]. This is sufficient to produce bistable circuits, but other mechanisms have also been shown [Keller, 1995; Siegal-Gaskins et al., 2011]. In particular, ultrasensitivity can arise by sequestration of a species [Buchler and Louis, 2008], like the way in which CI acts to sequester Tum in the MFL; as Tum levels increase, they are buffered by the titrant (CI), and thus a corresponding ultrasensitive response is observed to occur. The critical relation is the ultrasensitive decrease in free CI concentration as a function of the increase in production of Tum. That is, a change in the production rate of Tum results in an equivalently larger reduction in free CI in the ultrasensitive region.

With the complex requirements for balancing production rates and ultrasensitivity in the MFL, a mathematical model of this system will become indispensible, as it does for most projects in genetic network design [Kaern et al., 2003].

2.2 Developing a mathematical model of the Tum−CI MFL

The mathematical framework developed by François and Hakim [2005] to describe a MFL provides a good starting point for understanding the basic requirements for bistable or oscillating MFLs. Their analysis is based on a generic model of deterministic rate equations that characterise the time evolution of each species. In order to simultaneously model both activators and repressors, François and Hakim included terms for production from P_{B} (refer to Figure 2.1) in both bound and unbound states of the transcription factor A. A major conclusion of their work was that bistability was possible as long as the overall production rate for the unbound form of promoter P_{B} was higher than the rate of production from P_{A} and the overall production rate for the bound form of P_{B} was lower than the rate of production from P_{A}. They further showed that in this configuration the circuit can show history-dependent behaviour (hysteresis) as a function of the rate of production from P_{A}, a characteristic feature of bistable systems [Ferrell, 2002]. Here a similar model is employed, albeit with some minor modifications given the specific application to a Tum−CI MFL. In particular the Tum−CI and CI−*pR* interactions are both known to be highly cooperative, so here a model is formulated accordingly to gauge the effects of additional cooperativity on the bistability of the circuit.

The key network interactions of the Tum−CI MFL are the repression of *pR* by CI and the sequestration of CI due to the formation of the Tum−CI complex. Both of these can be thought of as equilibrium reactions that are characterised by the concentrations of free CI (specified as C), free Tum (specified as T) and Tum−CI complex (specified as S). The level of free CI determines the equilibrium between bound (specified as R for 'repressed') and

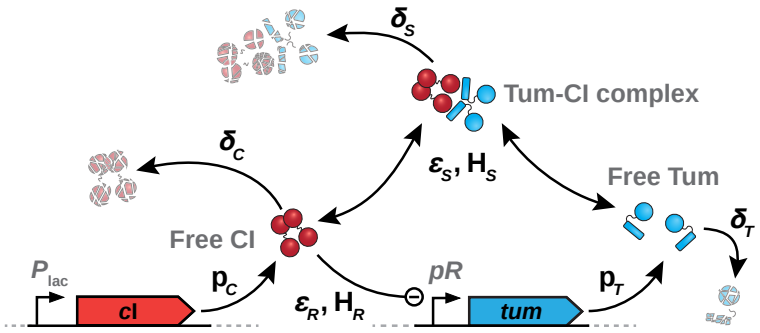


Figure 2.3: Parameters used for modelling the bistable Tum−CI MFL. The state of the system is defined by the concentrations of the three species illustrated (free Tum, free CI and Tum−CI complex). The time evolution of these species is defined by the production (p_C, p_T) and degradation (δ_C, δ_T, δ_S) rates and the network interactions by equilibria, which are approximately described using Hill curves with EC_{50}s (ε_R, ε_S) and Hill coefficients (H_R, H_S).

unbound (specified as *U* for 'unrepressed') forms of the *pR* promoter, but is itself determined by the equilibrium with free Tum and Tum−CI complex. A summary diagram of these reactions and the production and degradation reactions that constitute the Tum−CI MFL is shown in Figure 2.3. Each reaction is labelled by the parameters used to describe it, with subscripts chosen to reflect the species each parameter pertains to. The assumptions made in choosing this model and the corresponding set of parameters will be discussed and justified in the following sections, but put simply:

- the overall rate of production of CI is set by parameter p_C, with the overall rate of production of Tum from unbound *pR* set by p_T,
- degradation/loss of each species is assumed to be first order in the concentration of that species with degradation rates δ_C, δ_T and δ_S for free CI, free Tum and Tum−CI respectively, and
- the network interactions are described using Hill approximations for the equilibria with Hill coefficients H_R and H_S, and EC_{50}s ε_R and ε_S for the CI−*pR* and Tum−CI equilibria respectively.

It is worth noting here that a significant departure from the François and Hakim model is to neglect modelling of the intermediate mRNA species in the production of Tum. The delay introduced by the mRNA half-life was critical for getting oscillations in their model [François and Hakim, 2005]. However, here only the bistable mode is of interest so such dynamics are less relevant. This approximation simplifies the steady-state analysis and does not significantly impact the relevant deterministic dynamics as long as the half-life of the mRNA is much shorter than that of the protein [Sneppen et al., 2010]. This same assumption was successfully applied in the simple mathematical model used to describe the first synthetic bistable circuit [Gardner et al., 2000].

2.2.1 Modelling the CI−*pR* interaction

Of the two equilibrium reactions, repression of *pR* by CI is better understood. The available experimental data suggests that CI forms dimers at very low concentrations — with a dissociation constant of at most 1×10^{-8} M [Shearwin and Egan, 1996] — and that these CI dimers can bind cooperatively to form a heptamer (7-mer) of dimers [Pinkett et al., 2006]. There are a number of sites in the phage where CI is able to bind, including the opposing promoters *pR* and *pL* and two distant flanking sites, F_{R} and F_{L}, that increase the level of repression at *pR* and *pL* [Dodd and Egan, 2002]. Modelling of the *in vivo* repression data shows that CI binds cooperatively to repress the *pR* promoter and further suggested that the DNA wraps around the CI 14-mer, forming a CI wheel structure [Dodd et al., 2007b]. DNA wrapping of the 14-mer and DNA looping of the flanking sites have both been confirmed *in vitro* by atomic force microscopy and tethered particle motion [Wang et al., 2013]. For the Tum−CI MFL, the *pR* promoter *without* the flanking binding sites was chosen for its balance between ultrasensitivity and dynamic range.

Here, a model for the CI−*pR* equilibrium similar to that used by Dodd et al. [2007b] is employed to characterise the cooperative equilibrium between bound and unbound CI at the *pR* promoter. In this model, the complex series of reactions leading from CI monomers to the complete 14-mer are approximated (1) by presuming that the only CI species capable of binding to and repressing the *pR* promoter is the complete CI wheel, and (2) by using a Hill approximation to ignore all intermediate states (for which little data is available). It is further assumed here that at physiologically relevant CI protein concentrations, CI exists almost completely as dimers, like for the phage λ model [Ackers et al., 1982]. As a result the action of CI at *pR* can be modelled by a single equilibrium reaction between the unbound *pR* promoter and free CI dimers and the repressed promoter bound by a CI wheel ($\mathrm{CI}_{14}-pR$):

$$7\,\mathrm{CI}_2 + pR \rightleftharpoons \mathrm{CI}_{14}-pR \tag{2.1}$$

According to this formula, the concentration, R, of repressed *pR* and concentration, U, of unrepressed *pR* are related by equilibrium considerations to the concentration, C, of free CI as:

$$R = U\left(\frac{C}{\varepsilon_R}\right)^{\mathrm{H}_R} \tag{2.2}$$

with reaction order (Hill coefficient) $\mathrm{H}_R$and EC_{50} ε_R. According to this description of promoter binding, the fraction of unbound *pR*, f_U, can be written as a Hill equation:

$$f_U = \frac{U}{U+R} = \frac{U}{U+U(C/\varepsilon_R)^{\mathrm{H}_R}} = \frac{1}{1+(C/\varepsilon_R)^{\mathrm{H}_R}} \tag{2.3}$$

and conversely the fraction of bound *pR*, f_R, can be written:

$$f_R = \frac{R}{U+R} = \frac{1}{1+(C/\varepsilon_R)^{-\mathrm{H}_R}} \tag{2.4}$$

To relate this CI–DNA binding data to a production rate from pR, it is assumed here that production from the pR promoter is negligible when CI is bound, so that the promoter firing rate is proportional to the fraction of unbound pR, as given in Equation (2.3). Note that this is a gross oversimplification in comparison with the detailed model developed by Dodd et al. [2007b], which additionally accounts for a number of other unproductive states and also the effects of transcriptional interference at pR due to firing from pL. However, the simple proportionality model results in a far more tractable analysis, whilst still capturing the essential behaviour for the pR promoter [2], and has been successfully applied to describe other bistable gene networks [Gardner et al., 2000; Ozbudak et al., 2004]. In effect, the complexity is hidden within a fixed constant of proportionality, p_T, which can be set by measuring pR activity when $C = 0$, that is, in the absence of CI.

Using a DNA fragment containing the entire pR–pL region without flanking binding sites, the fraction of bound DNA has been measured *in vitro* as a function of CI concentration by gel shift assay [Dodd and Egan, 1996]. Using that data, the parameters in Equation (2.4) were fit to give a Hill coefficient of $\mathrm{H}_R = 1.7$ and an EC_{50} of $\varepsilon_R = 28$ nM [Shearwin et al., 1998]. This is in contrast with the parameters determined by Dodd et al. [2007b] to fit the *in vivo* repression data, where the Hill coefficient was determined as $\mathrm{H}_R = 2.2$ and the EC_{50} as $\varepsilon_R = 0.57$ WLU (wild-type lysogenic units) [Dodd et al., 2007b], which is equivalent to 630 nM using the quantitative Western blot measurements of Dodd and Egan [2002]. Perhaps most surprising is the order of magnitude difference in EC_{50} between *in vitro* and *in vivo* data sets. This seeming discrepancy can be partly rationalised on the basis that there is a much higher abundance of non-specific DNA-binding sites *in vivo*, thus increasing the effective concentration of CI required to repress pR. Note that for the models presented here, where it is assumed that the atomic CI unit is the dimer, the EC_{50}s reported above should be halved to give 14 nM worth of dimers *in vitro* and 315 nM *in vivo* with negligible changes to the Hill coefficients.

2.2.2 Modelling the Tum−CI interaction

The antirepressor Tum inactivates CI by suppressing its DNA-binding activity and is thus presumed to involve a direct protein-protein interaction [Shearwin et al., 1998]. The equilibrium has been characterised *in vitro* using a gel-shift assay to measure the decrease in the fraction of DNA bound by CI as a function of increasing Tum concentration. For a total CI concentration of $C_{\mathrm{tot}} = 150$ nM, the response showed high cooperativity with a Hill coefficient of 4.5 and half maximal inhibitory concentration (IC_{50}) of 3.3 μM. The fraction of DNA bound by CI is an indirect measure of Tum activity, but by using the *in vitro*CI−pR binding data discussed in the previous section, the Tum−CI interaction can be more directly examined as a change in concentration of free CI as a function of increasing Tum.

[2] The effect of transcriptional interference on pL is far more pronounced, and further, the pR promoter has three strong repressor binding sites, whilst the pL promoter has only one.

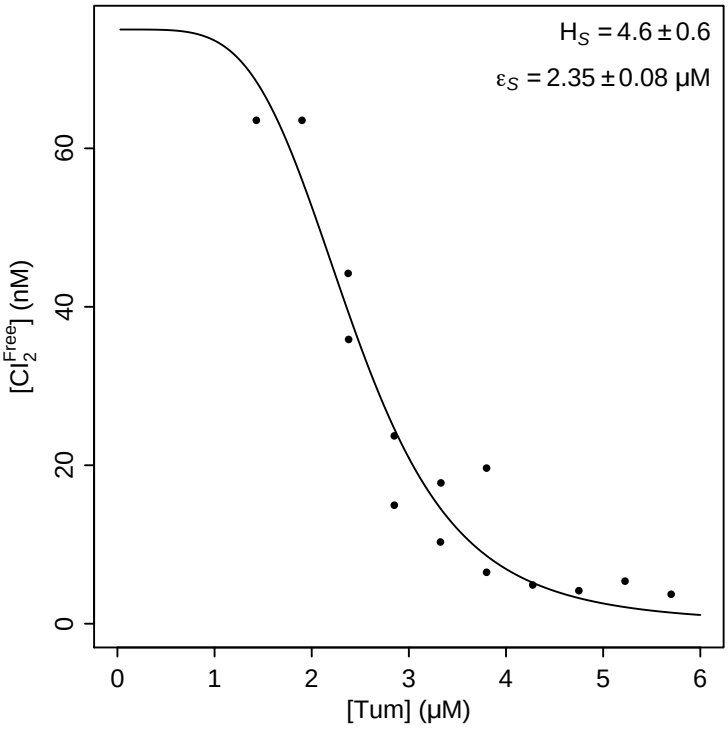


Figure 2.4: The Tum−CI equilibrium model derived in the text (solid) provides a good fit for the *in vitro* gel-shift data (points) of Shearwin et al. [1998]. The gel-shift assay was used to quantify the decrease in the fraction of DNA bound by CI for increasing concentrations of Tum, but the original data has been transformed here to the concentration of free CI using a complementary gel shift measurement that characterised the CI−*pR* interaction alone. Listed at top right with standard errors in the fit, are the Hill coefficient, H_S, and EC_{50}, ε_S, determined for the Tum−CI equilibrium model presented in the text.

This is achieved by rearranging Equation (2.4) in terms of the fraction, f_R, of bound DNA:

$$C = \varepsilon_R \left(\frac{1}{f_R} - 1\right)^{-1/H_R} \tag{2.5}$$

and applying that equation to the data using the *in vitro* Hill coefficient and EC_{50} discussed in Section 2.2.1 (recall that $\varepsilon_R = 14$ nM worth of *dimers*). The data thus transformed is displayed in Figure 2.4, and shows the highly cooperative nature of the interaction. Note that 'free CI' here refers to the subset of CI that is neither bound by Tum nor bound to DNA, but the subset bound to DNA can be ignored since the concentration of DNA was presumed negligible in the experiment.

There are a number of alternative mechanisms by which Tum might prevent CI binding to DNA, including direct binding of Tum to the DNA-binding region of CI or disruption of CI multimerisation [Shearwin et al., 1998]. However, with relatively limited biochemical information available on the form of the Tum−CI interaction, a simple representative reaction scheme has been chosen:

$$2\,\mathrm{Tum} + \mathrm{CI}_2 \rightleftharpoons \mathrm{Tum}_2\mathrm{-CI}_2 \tag{2.6}$$

The stoichiometry has been chosen to reflect the significant cooperativity observed in response to changes in Tum concentration, and also to match what little is known about Tum multimerisation. Sedimentation equilibrium experiments have indicated that at a concentration of 9.5 μM, Tum acts like a single monomeric species with molecular weight twice that of a Tum monomer [Shearwin et al., 1998]. That is, at a concentration of 9.5 μM, Tum is

predominantly dimeric and, whilst unlikely, all ultrasensitivity in the present model of the Tum−CI interaction is attributed to this dimerisation. This scheme results in the following Tum−CI equilibrium equation relating the concentrations of free CI (C), free Tum (T), and Tum−CI complex (S):

$$S = \varepsilon_S^{-a-b+1} C^a T^b \tag{2.7}$$

where the concentrations C and T obey yet-to-be-determined reaction orders, a and b, and the coefficient ε_S is defined to be analagous to an EC_{50} with dimensions of unit concentration, hence the choice of the power for that term.

The *in vitro*Tum−CI gel shift data shows the response of free CI to the *total* concentration of Tum. However, the level of free Tum can be obtained from the total concentrations of Tum (T_{tot}) and CI (C_{tot}) using the mass-balance equations:

$$C_{\text{tot}} = C + S \tag{2.8a}$$

$$T_{\text{tot}} = T + 2S \tag{2.8b}$$

where it has been assumed that $C + S \gg 7R$, the concentration of CI bound to DNA, since the concentration of DNA is presumed negligible. This greatly simplifies the analysis since it means the Tum−CI and CI−pR equilibria operate independently of each other, and it also eliminates the need for a constraint equation for the total concentration of DNA. For known concentrations C_{tot} and T_{tot} and after substituting with Equation (2.7), the constraints in Equation (2.8) are a system of equations in two variables. By eliminating S from Equations (2.8a) and (2.8b), solving for T, and substituting this expression into Equation (2.7), Equation (2.8a) can be specified completely in terms of C:

$$C_{\text{tot}} = C + \varepsilon_S^{-a-b+1} C^a (T_{\text{tot}} - 2C_{\text{tot}} + 2C)^b \tag{2.9}$$

The only free variable in Equation (2.9) is C, which can be found using numerical root-finding algorithms. The non-linear least squares fitting algorithm of the R application [R Development Core Team, 2012] was used to fit the gel-shift data using this model with free parameters ε_S, a and b and with C_{tot} set to 75 nM worth of dimers. With all parameters left free the data was fitted well, but the parameter estimates were obtained with very low confidence. This was mainly due to a high correlation between the two power parameters, a and b. Hence, without any evidence to the contrary and since $b > a$ for the general fit, a was then fixed to 1 so that all cooperativity was attributed to Tum. This choice of parameters produced a similarly good fit of the data (as judged by the residual sum of squares), and is shown in Figure 2.4 on page 21. With this choice of reaction order, the equilibrium equation can be simplified as:

$$S = C \left(\frac{T}{\varepsilon_S} \right)^{\text{H}_S} \tag{2.10}$$

where $\text{H}_S = a$ has become the Hill coefficient for the Tum−CI equilibrium. Observe in the figure that since C_{tot} is small compared with T_{tot}, the concentration of S is also necessarily

small compared with T_{tot} to satisfy the mass balance equations. Hence, $T_{\text{tot}} \approx T$ and the value of ε_S (2.35 μM) is close to the EC_{50} of the displayed curve.

2.2.3 Deterministic free species model

The CI−*pR* and Tum−CI equilibria set the concentrations of the free CI and Tum species at steady state, but in the *in vivo* MFL these species additionally evolve over time according to protein production and degradation rates. To describe the complete gene network, then, the dynamics of the interaction equilibria need to be considered in the context of the dynamic production and degradation reactions. Such a time-dependent deterministic description of the Tum−CI MFL model can be fully represented as a set of first order ODEs in terms of the free species variables. To start with, the time evolution of the Tum−CI complex is derived. The concentration of the Tum−CI complex, S, changes in response to formation and dissociation of the complex according to the equilibrium established in Section 2.2.2, and is also lost *in vivo* by degradation. The differential equation for evolution of the complex reads:

$$\frac{\mathrm{d}S}{\mathrm{d}t} = -\left(\gamma + \delta_S + \delta_{SC} + \delta_{ST}\right) S + \mathrm{k}_S C T^{\mathrm{H}_S} - \mathrm{k}_{-S} S \tag{2.11}$$

For complete generality, degradation of the complex as specified in the above equation has been split into a number of terms: (1) a complex-specific degradation term, $-\delta_S S$, representing targeted degradation of the complex, (2) CI- and Tum-specific degradation terms, $-\delta_{SC} S$ and $-\delta_{ST} S$, representing targeted degradation (removal) of either CI or Tum respectively from the complex, and (3) a general growth-rate dependent dilution term, $-\gamma S$, that represents loss in complex concentration as the cells grow in volume. These terms sum together to give a bulk term for loss of the complex by degradation/dilution as shown. The degradation rates δ_{SC} and δ_{ST} reflect the level of protection that the complex affords to each species compared with active degradation of the free species.

The final two terms in Equation (2.11) describe evolution due to the Tum−CI equilibrium reaction. Here, however, the steady-state description derived in Section 2.2.2 has been split into separate rate law terms for formation and dissociation of the complex. The form of these terms is not immediately obvious and deserves some additional discussion. Rate law descriptions of typical one-step two-species binding equilibria can be neatly split between forwards and reverse reactions, with complex formation being a second-order term in the two reactant concentrations and complex dissociation being a first-order term in product concentration. That is, $\frac{\mathrm{d}S}{\mathrm{d}t} = \mathrm{k}_S C T - \mathrm{k}_{-S} S$, where the on rate, k_S, and off rate, k_{-S}, are set such that at equilibrium ($\frac{\mathrm{d}S}{\mathrm{d}t} = 0$) their ratio gives the association constant, K, for the reaction ($K = \mathrm{k}_S/\mathrm{k}_{-S}$). With such a simple scheme, the association constant is merely the inverse of the EC_{50}, so that $\varepsilon_S = 1/K$. The off rate (with units of inverse time) effectively sets the timescale of the reaction, that is, the time it takes to reach equilibrium. This is made more transparent when the differential equation (for one-step complex

formation) is rewritten:

$$\frac{\mathrm{d}S}{\mathrm{d}t} = \mathrm{k}_{-S}\left(KCT - S\right) \tag{2.12}$$

Clearly, the association constant, K, simply sets the steady-state species concentrations (consider the steady state, that is, when $\frac{\mathrm{d}S}{\mathrm{d}t} = 0$), and k_{-S} scales the time taken to get there.

The model of the Tum−CI interaction in Section 2.2.2 accounts for a more complicated multi-step reaction scheme by simplifying it to an effective one-step reaction using a Hill approximation, where the simple one-step two-species intermediate reactions are neglected. Such an approximation reduces the complexity and number of parameters required to describe the equilibrium, but makes it more difficult to interpret the meaning of bulk on and off rates for what is in fact a multi-step reaction. Nonetheless, to retain the simplicity of a one-step reaction in the deterministic model, the Tum−CI interaction is split into reactant and product terms as given on either side of the equilibrium relation in Equation (2.10) (Section 2.2.2), and the differential equation for time evolution of the complex is rewritten:

$$\frac{\mathrm{d}S}{\mathrm{d}t} = -\left(\gamma + \delta_S + \delta_{SC} + \delta_{ST}\right)S + \mathrm{k}_{-S}C\left(\frac{T}{\varepsilon_S}\right)^{\mathrm{H}_S} - \mathrm{k}_{-S}S \tag{2.13}$$

The off rate, k_{-S}, is used as a multiplier for both directions of the binding reaction and hence sets the timescale of the reaction as it would in the case of the two-species binding equilibrium described by Equation (2.12). Furthermore, the EC_{50} for the Tum−CI reaction is used instead of the association constant since it is easier to interpret within the Hill approximation and hence serves as a more useful parameterisation. Observe that if degradation of the complex is ignored in Equation (2.13), then at steady state the equilibrium equation (Equation (2.10)) is obtained.

The time evolution of the free CI dimers is now considered. The pool of free CI dimers with concentration C participate in both the Tum−CI and CI−pR equilibria and are further subject to degradation and dilution that are complemented by production from the P_{lac} promoter. Hence, the time evolution of free CI dimers is modelled by the following differential equation:

$$\frac{\mathrm{d}C}{\mathrm{d}t} = \mathrm{p}_C - \delta_C\mathrm{C} + \delta_{ST}S - \mathrm{k}_{-S}\mathrm{C}\left(\frac{T}{\varepsilon_S}\right)^{\mathrm{H}_S} + \mathrm{k}_{-S}S \tag{2.14}$$

The steps involved in production of CI from P_{lac} have been rolled into a single zeroth order production term p_C. The first order terms for degradation and dilution have been combined together as a single constant δ_C, which for convenience has been defined as the sum of the general species dilution rate, γ, and the rate of active degradation, so that the active degradation rate of CI is given by $\delta_C - \gamma$. Degradation of Tum from the Tum−CI complex leads to increased availability of free CI as given by the $\delta_{ST}S$ term. The Tum−CI formation and dissociation terms are simply the reverse of those formulated for the rate of change of S in Equation (2.13). Finally, whilst CI participates in the CI−pR equilibrium,

here it is assumed that the concentration of *pR* binding sites *in vivo* will be negligible in comparison with the concentrations of CI that occur within a functional bistable MFL, and hence have little effect on the dynamics of free CI.

Finally, the time evolution of free Tum can be considered. The production rate of Tum from *pR* can be assumed proportional to the concentration (availability) of unrepressed *pR* promoters, U, providing a one step model of production is used. Production and degradation typically occur on a much slower timescale than binding equilibria, so by further assuming that the dynamics of CI binding to *pR* are always at equilibrium, then, as discussed in Section 2.2.1, the production rate of Tum from *pR* can be written as a function of free CI concentration, that is, production of Tum is proportional to the fraction of unrepressed *pR* given by Equation (2.3) on page 19. Hence the following differential equation for the evolution of free Tum:

$$\frac{\mathrm{d}T}{\mathrm{d}t} = \frac{\mathrm{p}_T}{1+\left(\frac{C}{\varepsilon_R}\right)^{\mathrm{H}_R}} - \delta_T T + 2\delta_{SC} S - 2\mathrm{k}_{-S} C \left(\frac{T}{\varepsilon_S}\right)^{\mathrm{H}_S} + 2\mathrm{k}_{-S} S \tag{2.15}$$

As done for the free CI equation, degradation and dilution of Tum are combined into a single degradation rate, δ_T. Degradation of CI in the complex leads to an increase in free Tum, but the reaction stoichiometry suggested in Section 2.2.2 was that the Tum−CI complex was formed from one CI dimer and two Tum monomers, hence one unit of CI (dimers) and two units of Tum are returned for every unit of the complex ($2\delta_{SC}S$). This similarly applies for the Tum−CI equilibrium terms which are identical to those for the free CI equation apart from the factor of two.

Putting these equations together, the system of first order ODEs governing the concentrations of CI (C), Tum (T) and Tum−CI complex (S) in the Tum−CI MFL are written as:

$$\frac{\mathrm{d}C}{\mathrm{d}t} = \mathrm{p}_C - \delta_C C + \delta_{ST} S - \mathrm{k}_{-S} C \left(\frac{T}{\varepsilon_S}\right)^{\mathrm{H}_S} + \mathrm{k}_{-S} S \tag{2.16a}$$

$$\frac{\mathrm{d}T}{\mathrm{d}t} = \frac{\mathrm{p}_T}{1+\left(\frac{C}{\varepsilon_R}\right)^{\mathrm{H}_R}} - \delta_T T + 2\delta_{SC} S - 2\mathrm{k}_{-S} C \left(\frac{T}{\varepsilon_S}\right)^{\mathrm{H}_S} + 2\mathrm{k}_{-S} S \tag{2.16b}$$

$$\frac{\mathrm{d}S}{\mathrm{d}t} = -\left(\delta_S + \gamma + \delta_{SC} + \delta_{ST}\right) S + \mathrm{k}_{-S} C \left(\frac{T}{\varepsilon_S}\right)^{\mathrm{H}_S} - \mathrm{k}_{-S} S \tag{2.16c}$$

Note that in contrast with the MFL model developed by François and Hakim [2005], here the equilibrium between free and sequestered Tum and CI is explicitly modelled. François and Hakim chose to ignore feedback resulting from dissociation of the complex, essentially by employing a pseudo-equilibrium assumption for evolution of the complex. This is equivalent to setting $\frac{\mathrm{d}S}{\mathrm{d}t} = 0$ in the above equations, which leads to substantial simplifications. This assumption relies on the complex being a transient population that is actively degraded. In the case of the Tum−CI interaction, on the basis that Tum can act

reversibly against CI *in vitro* without the need for degradation [Shearwin et al., 1998], and without any evidence for targeted degradation *in vivo*, it is likely that Tum acts as a molecular 'sink' for CI, with sequestration being the primary mechanism of action. Hence the effects of 'retroactivity' — where the consumption of an active species (either Tum or CI in this case) feeds back on system dynamics [Del Vecchio et al., 2008] — cannot be ignored, and the pseudo-equilibrium assumption is not valid in this case. Note that whilst pseudo equilibrium was assumed for the CI$-pR$ interaction, this can be justified on the basis that the concentration of CI binding sites at pR are low compared with the relevant concentrations of free CI. The same cannot be said of the Tum$-$CI interaction, since in that case the concentration of the complex cannot be assumed small compared with the concentrations of free CI and Tum.

The ODEs as written have a somewhat overcomplicated parameterisation of degradation. Without evidence for any protection from degradation of CI and Tum when part of the Tum$-$CI complex, it seems reasonable at this stage to assume that degradation of CI and Tum when in the complex simply occurs at the same rate as active degradation of the free species, that is, $\delta_{SC} = \delta_C - \gamma$, and $\delta_{ST} = \delta_T - \gamma$. This is the same as assuming that neither species receives any protection from degradation when part of the complex. The result of this minor modification is a reduced number of parameters in Equation (2.16) so that the system of equations can be rewritten:

$$\frac{\mathrm{d}C}{\mathrm{d}t} = \mathrm{p}_C - \delta_C C - \mathrm{k}_{-S} C \left(\frac{T}{\varepsilon_S}\right)^{\mathrm{H}_S} + \left(\delta_T - \gamma + \mathrm{k}_{-S}\right) S \tag{2.17a}$$

$$\frac{\mathrm{d}T}{\mathrm{d}t} = \frac{\mathrm{p}_T}{1 + \left(\frac{C}{\varepsilon_R}\right)^{\mathrm{H}_R}} - \delta_T T - 2\mathrm{k}_{-S} C \left(\frac{T}{\varepsilon_S}\right)^{\mathrm{H}_S} + 2\left(\delta_C - \gamma + \mathrm{k}_{-S}\right) S \tag{2.17b}$$

$$\frac{\mathrm{d}S}{\mathrm{d}t} = \mathrm{k}_{-S} C \left(\frac{T}{\varepsilon_S}\right)^{\mathrm{H}_S} - \left(\delta_S + \delta_C + \delta_T - \gamma + \mathrm{k}_{-S}\right) S \tag{2.17c}$$

2.2.4 Deterministic total species model

Whilst the model in terms of the free species and Tum$-$CI complex is sufficient to completely describe the system, the form of the equations is simplified and the origin of bistability easier to understand when they are written in terms of the evolution of the total concentrations of Tum and CI. Nonetheless, the system of ODEs in terms of the free species is still easier to solve numerically, so those equations will remain useful. A system of ODEs in terms of the total concentrations of Tum and CI can be obtained by differentiating the mass balance equations given in Equations (2.8a) and (2.8b) by time, and substituting from

Equations (2.8) and (2.17) to give:

$$\frac{\mathrm{d}C_{\mathrm{tot}}}{\mathrm{d}t} = \mathrm{p}_C - \delta_C C_{\mathrm{tot}} - \delta_S S \tag{2.18a}$$

$$\frac{\mathrm{d}T_{\mathrm{tot}}}{\mathrm{d}t} = \frac{\mathrm{p}_T}{1 + \left(\frac{C}{\varepsilon_R}\right)^{\mathrm{H}_R}} - \delta_T T_{\mathrm{tot}} - 2\delta_S S \tag{2.18b}$$

This does not eliminate the dependence on the free species, but if it is assumed that the fast binding reactions, that is, the Tum−CI and CI−*pR* equilibria, are always at equilibrium as constrained by the time-varying total concentrations of CI and Tum, then the free species variables can instead be considered as (somewhat complicated) functions of C_{tot} and T_{tot}. With the mass balance equations as given (i.e., neglecting CI bound at *pR*), then the concentration of free CI is simply found by solving for C in Equation (2.9) as was done for the *in vitro*Tum−CI equilibrium. For reference, the constraint equation reads:

$$C_{\mathrm{tot}} = C + C\left(\frac{T_{\mathrm{tot}} - 2C_{\mathrm{tot}} + 2C}{\varepsilon_S}\right)^{\mathrm{H}_S} \tag{2.19}$$

which cannot be solved in terms of C analytically, but can be solved using numerical root-finding procedures. The concentration of the complex, S, can then simply be deduced from the mass balance equations as $S = C_{\mathrm{tot}} - C$. Note that this equilibrium assumption is *not* the same as the pseudo-equilibrium assumption used by François and Hakim [2005] and discussed in the previous section, since time evolution of the complex is still accounted for as part of the total species concentrations. Loss of the complex by dilution and by degradation at the standard free species degradation rates is implicit in the total species ODEs. The total species degradation terms become more complicated if either of the species are protected from degradation when bound up in the Tum−CI complex. In that case, there would be additional dependence in the degradation rate terms on the concentration of the complex.

If targeted degradation of the complex is neglected, that is, $\delta_S = 0$, then the form of the equations becomes particularly simple and the origin of bistability in this model of the Tum−CI MFL is made transparent. With a constant production rate of CI, Equation (2.18a) tends towards a single steady-state concentration of total CI. So like with the *in vitro* gel shift assay of the Tum−CI interaction discussed in Section 2.2.2, the total CI concentration essentially remains fixed regardless of Tum concentration. On the other hand, the differential equation for total Tum, Equation (2.18b), does depend on the Tum−CI equilibrium. The steady states for that equation occur when the production rate is equal to the degradation rate. These terms are shown plotted against the concentration of total Tum in Figure 2.5. Bistability arises as a result of the sigmoidal shape of the production rate of Tum from *pR*. The shape of that curve is a direct result of the combined CI−*pR* and Tum−CI equilibria, and is in fact proportional to the curve expected for an *in vitro* measurement of the *unbound* fraction of DNA in the Tum−CI gel shift experiment (see Section 2.2.2.

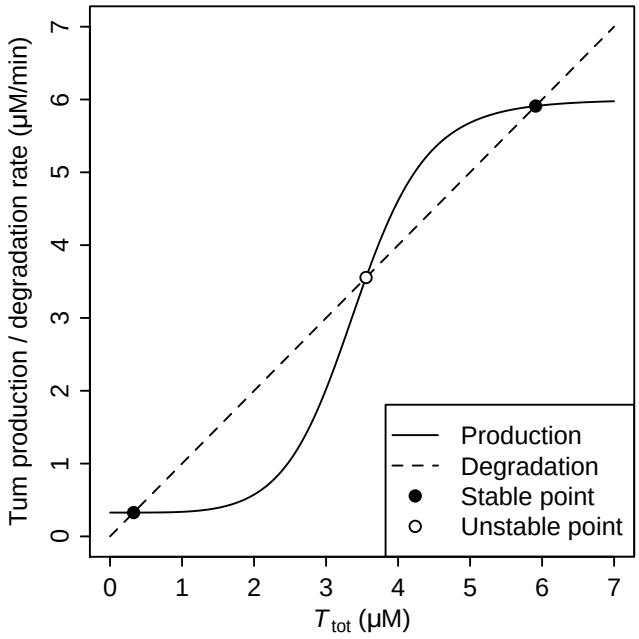


Figure 2.5: Bistability in the Tum−CI MFL arises as a result of the sigmoidal response of Tum production as a function of total Tum concentration. Shown plotted are the production and degradation rate curves from the differential rate equation for total Tum. The production rate curve is equivalent to the response of the unbound fraction of *pR* DNA in the *in vitro*Tum−CI gel shift assay as a function of total Tum concentration (compare with Figure 2.4). For the sake of example, parameters were chosen that match the *in vitro* equilibria ($\varepsilon_R = 14$ nM, $\mathrm{H}_R = 1.7$, $\varepsilon_S = 2.35$ μM, $\mathrm{H}_S = 4.6$) where the steady-state concentration of total CI has also been set at $C_{\mathrm{tot}} = 75$ nM worth of dimers. To balance production and degradation, p_T was set to 6 μM/min and δ_T to 1 min^{-1}. The filled and unfilled circles mark the two stable and one unstable equilibrium points respectively.

When the total Tum concentration is zero, the level of *pR* repression is set by that level expected when $C = C_{\mathrm{tot}}$, the steady-state concentration of total CI, hence the non-zero offset at $T_{\mathrm{tot}} = 0$. At this end of the curve, production outweighs degradation, and Tum concentration will tend to increase to the low T_{tot} stable equilibrium point. At the other end of the curve, $T_{\mathrm{tot}}$is high enough such that repression of the promoter is completely relieved, and *pR* production reaches its maximum rate. However, at these concentrations, degradation outweighs production, so total Tum concentration will tend to decrease to the high T_{tot} equilibrium where production and degradation are balanced. Between these two stable points, an unstable equilibrium point exists; any deviation from that point results in the system tending towards one of the stable points.

As a final note, an important (and easily modified) parameter is the production rate of CI, which sets the steady-state concentration of CI, C_{tot}. Variations in C_{tot} affect the shape of the Tum−CI equilibrium curve, and hence the shape of the production rate of Tum from *pR*. Observe that as $\mathrm{p}_C \to 0$, then $C_{\mathrm{tot}} \to 0$, and hence by the mass balance equations, $C \to 0$. So in this case the system of ODEs becomes monostable with a single high T_{tot} state. Conversely as $\mathrm{p}_C \to \infty$, then $C_{\mathrm{tot}} \to \infty$, and when $C_{\mathrm{tot}} \gg T_{\mathrm{tot}}$ then $C \to \infty$, and the term for Tum production vanishes so that the MFL becomes monostable with a low T_{tot} state. Between these two extremes, however, a subset of steady-state CI concentratios that are permissive for bistability can be anticipated.

2.3 Steady-state analysis of the Tum−CI MFL model

For the Tum−CI MFL to exhibit bistability, the system of ODEs by which it is characterised must produce two stable fixed points at steady state. An analysis of the steady-state equilibrium points can thus reveal the parameter values that support bistability. In this section, a general formula that can be solved to find the fixed points of the Tum−CI MFL is derived from the free species ODEs, and is then applied to determine the regions of bistability as a function of the model parameters.

2.3.1 Solving the free species model at steady-state

The steady-state fixed points, whether derived from the total or free species ODEs, should be equivalent, but at this point it is easier to start the analysis with the system of ODEs in terms of free species. At steady state, the rate of change of each species vanishes such that production and degradation are balanced. Setting the rates of change to zero in the system of ODEs derived in Section 2.2.3 produces three equations in terms of three variables, C, T and S, which can be solved in terms of those variables to find the fixed points of the Tum−CI MFL.

To begin with, just the equation for $\frac{\mathrm{d}S}{\mathrm{d}t}$ is set to zero; this constitutes the case of a quasi-equilibrium assumption and the formulation made in the François and Hakim model. With this assumption, Equation (2.17) can be rewritten:

$$\frac{\mathrm{d}C}{\mathrm{d}t} = \mathrm{p}_C - \delta_C C - (\delta_C + \delta_S) C \left(\frac{T}{\varepsilon_S}\right)^{\mathrm{H}_S} \tag{2.20a}$$

$$\frac{\mathrm{d}T}{\mathrm{d}t} = \frac{\mathrm{p}_T}{1 + \left(\frac{C}{\varepsilon_R}\right)^{\mathrm{H}_R}} - \delta_T T - 2(\delta_T + \delta_S) C \left(\frac{T}{\varepsilon_S}\right)^{\mathrm{H}_S} \tag{2.20b}$$

In the above equations, the use of ε_S is an approximation. It is more precisely defined as an effective EC_{50} for the Tum−CI equilibrium given by:

$$\left(\frac{1}{\varepsilon_S^{\mathrm{eff}}}\right)^{\mathrm{H}_S} = \frac{\mathrm{k}_{-S}}{\delta_S + \delta_C + \delta_T - \gamma + \mathrm{k}_{-S}} \left(\frac{1}{\varepsilon_S}\right)^{\mathrm{H}_S} \tag{2.21}$$

However, it is written as ε_S in the system of ODEs, since equilibration of the complex is expected to occur on a much faster timescale than production and degradation of the species. This means that k_{-S} will be much larger than the combined rates of degradation and dilution, so that, to a good approximation, the two EC_{50}s are equal.

Note the similarity of the quasi-equilibrium equations with the formulation in terms of total species (Equation (2.18)). Here, however, the free species T and C have to take over the role of the complex. So whilst the quasi-equilibrium formulation is much simpler than the formulations including modelling of the complex (that is, those based on either Equation 2.17 or 2.18), it performs quite differently when far from equilibrium.

Now setting $\frac{dC}{dt}$ and $\frac{dT}{dt}$ close to zero, the fixed points of Equation (2.20) can be determined. First take $\frac{dT}{dt} - 2\frac{\delta_T+\delta_S}{\delta_C+\delta_S}\frac{dC}{dt}$ and rewrite in terms of T to obtain:

$$\delta_T T = \frac{p_T}{1+(C/\varepsilon_R)^{H_R}} - 2\frac{\delta_T+\delta_S}{\delta_C+\delta_S}(p_C - \delta_C C) - \left(\frac{dT}{dt} - 2\frac{\delta_T+\delta_S}{\delta_C+\delta_S}\frac{dC}{dt}\right) \tag{2.22}$$

When close to steady state, the differentials in the above equation will be small and can be ignored as long as the steady-state concentration of free Tum scaled by its degradation rate is much larger than zero, that is, $\delta_T T \gg 0$. With that approximation, the free Tum equation can be substituted into Equation (2.20a) to obtain:

$$\frac{dC}{dt} \approx p_C - \delta_C C - \frac{(\delta_S+\delta_C)C}{\varepsilon_S^{H_S}}\left(\frac{p_T/\delta_T}{1+(C/\varepsilon_R)^{H_R}} - 2\frac{\delta_T+\delta_S}{\delta_T}\frac{\delta_C}{\delta_C+\delta_S}\left(\frac{p_C}{\delta_C} - C\right)\right)^{H_S}$$

which becomes exact as the system approaches steady state ($\frac{dC}{dt} = 0$). This approximates the system of three ODEs as a one-dimensional differential equation, which can be plotted against the concentration of free CI to judge the Jacobian by eye (the Jacobian here is just the slope of the above equation in terms of its only free variable, C). A more practical expression can be obtained using the following substitutions:

$$M_C = \frac{p_C}{\delta_C} \qquad D_C = \frac{\delta_C}{\delta_C+\delta_S}$$
$$M_T = \frac{p_T}{\delta_T} \qquad D_T = \frac{\delta_T}{\delta_T+\delta_S}$$

to write the steady-state equation in terms of maximal steady-state CI and Tum levels (M_C and M_T) and fractional degradation adjustments (D_C and D_T):

$$\frac{dC}{dt} \approx \delta_C\left(M_C - C - \frac{C}{D_C\varepsilon_S^{H_S}}\left(\frac{M_T}{1+(C/\varepsilon_R)^{H_R}} - 2\frac{D_C}{D_T}(M_C - C)\right)^{H_S}\right) \tag{2.23}$$

With the number of parameters thus reduced, then, aside from the parameters used to describe the equilibria, the only parameters that remain to be estimated are the maximal production rates and the fractional changes made to the degradation rates. To begin with, rough estimates for these parameters are made here; the effects of parameter adjustments starting from this first guess will be considered later.

Steady-state levels of CI produced from the P_{lac} promoter on a single-copy plasmid (pZC320) have been measured by Western blot for an IPTG-inducible system and ranged from undetectable when uninduced to 2390 CI monomers per cell with 100 μM IPTG [Dodd and Egan, 2002]. Using the concentration factor prescribed in the paper (1.27 nM) this gives a maximum steady-state level of CI *dimers* as $M_C \approx 1500$ nM. Since the production rate from P_{lac} can be set in an inducible manner, it stands as the easiest parameter to modify in the MFL. Hence, particular attention will be paid to variations in terms of this parameter, that is, CI production rate (p_C), or equivalently, the maximum concentration of total CI (M_C).

The unrepressed production rate of Tum from the *pR* promoter has not previously been measured, so an estimate will be made by extrapolation from that made for CI. An integrated P_{lac}–*lacZ* reporter construct was assayed in a strain with LacI supplied from the pUHA-1 plasmid at various concentrations of IPTG and measured to be approximately 250 LacZ units at 100 μM IPTG [Adam Palmer, unpublished data]. A similar integrated *pR*–*lacZ* reporter construct with consistent auxiliary plasmids produced an average unrepressed activity of approximately 860 LacZ units over a range of IPTG concentrations [Crooks, 2006]. This makes the *pR* promoter around 3.4 times stronger than P_{lac} at that level of induction. If it is assumed that translation and degradation rates of Tum match those of CI, and using the measured copy number of 1.4 ± 0.2 copies per chromosome for the pZC320 plasmid with mini-F origin [Shi and Biek, 1995], then the extrapolated steady-state level of Tum becomes $\mathrm{M}_T = (3.4 \text{ times stronger}) \times (1.4 \text{ fewer copies})^{-1} \times (3035 \text{ nM of monomers}) \approx 7400$ nM.

Without any reason to suspect a degradation-mediated Tum antirepressor activity, a reasonable first assumption is that there is no additional degradation of the complex, that is, $\delta_S = 0$ and hence $\mathrm{D}_C = \mathrm{D}_T = 1$. Parameters describing the CI−*pR* and Tum−CI equilibria were discussed in Sections 2.2.1 and 2.2.2, where specifically, the *in vivo*CI−*pR* parameters are used ($\mathrm{H}_R = 2.2$ and $\varepsilon_R = 315$ nM worth of CI *dimers*), whereas the *in vitro*Tum−CI parameters are used ($\mathrm{H}_S = 4.6$, $\varepsilon_S = 2350$ nM worth of Tum monomers). This should present a reasonable choice of parameters, since the increased occurrence of non-specific DNA binding *in vivo* is anticipated to affect the CI−*pR* equilibrium, whilst the Tum−CI interaction is anticipated to occur in solution and more closely match the *in vitro* parameterisation.

With these parameter choices, Equation (2.23) can be plotted as a function of free CI, as shown in Figure 2.6. Recall that the rate equation as plotted is only a valid approximation for points close to equilibrium, however, the plots assist in finding the roots of the rate equation and also in interpreting the stability of each root. The concentrations of free CI at which the rate equation intersects the x-axis are the steady-state concentrations. When the slope of the rate equation at the zero point is decreasing, that state is stable; if the concentration C deviates from that steady-state value, the rate of change returns the system to equilibrium. When the slope of the rate equation is increasing, however, that steady-state is unstable; any increase/decrease in C results in a positive/negative rate of change, taking the system away from equilibrium.

Since the maximum concentration of total CI can be set by changing the level of induction of the P_{lac} promoter, the rate equation curve is depicted in the figure for three different choices of this parameter. This reveals that for certain choices of M_C, the rate equation is monostable with a single equilibrium point ($\mathrm{M}_C = 0.5$ or 1.46 μM), but for $\mathrm{M}_C = 1.0$ μM, the rate equation is bistable with two stable states straddling a single unstable state. Shown overlaid on each plot is the concentration of free Tum that would occur

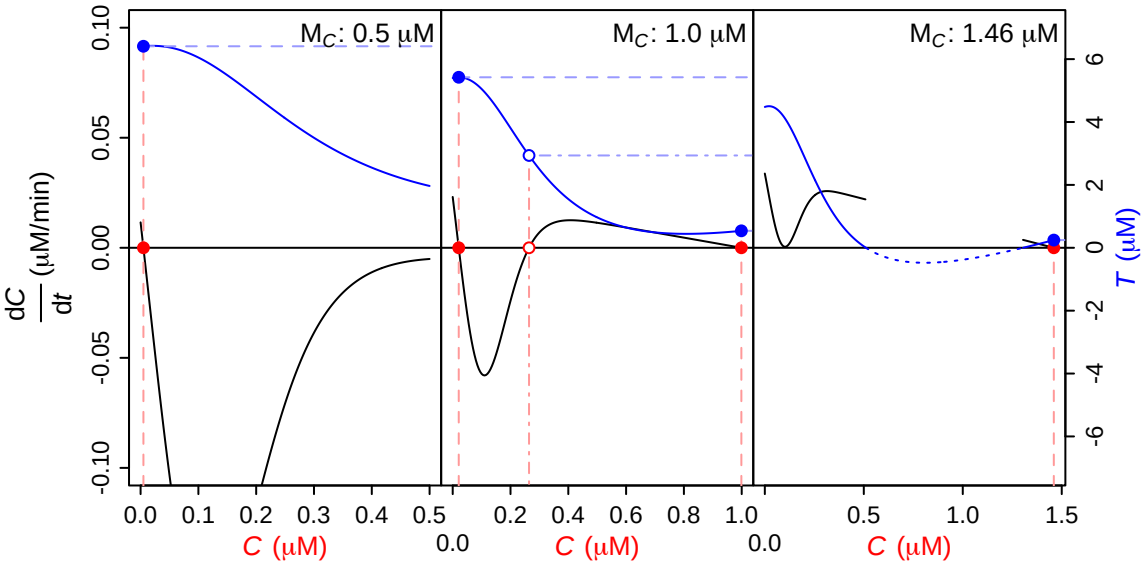


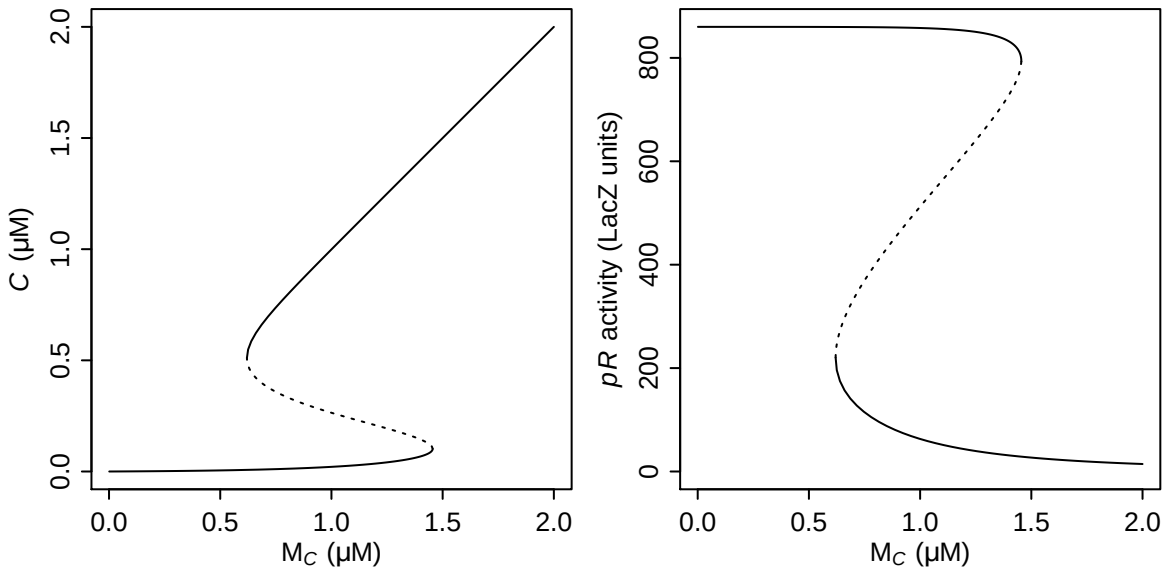
Figure 2.6: The Tum−CI MFL model supports bistability as characterised by the steady-state values of the system of ODEs in terms of free species concentrations. Shown plotted is the value of $\frac{dC}{dt}$, as given by a one-dimensional approximation to the full model that is only accurate for values of C close to equilibrium (as described in the text). The parameters used are those described in the text, though the existence of two stable points (bistability) requires an alternative choice of parameter, M_C, for maximum CI steady-state concentration, which is listed at the top right of each plot. Steady-state points are indicated by closed (stable) and open (unstable) circles and marked along the free CI, C, axis using dashed lines. Shown in blue is the value of free Tum, T, at each concentration of free CI, and the respective steady-state values for T are marked on the right axis.

for a given free CI concentration as calculated by Equation (2.22) (note this is still only accurate for concentrations close to the equilibrium points). As can be seen in Figure 2.6, the two stable points in the centre panel can be described alternatively as low C and high C, or high T and low T states respectively.

The value of M_C for which the system transitions from monostability to bistability is termed a bifurcation point, which are more generally the points in model systems where changes in parameter values cause a changes in stability. Observe that the rate equation curve for $M_C = 1.46$ μM is very close to the one of the bifurcation points in terms of the parameter M_C; the bifurcation occurs when the local minimum at low Cconcentrations touches the x-axis. The range of M_C concentrations that can support bistability must lie between two such bifurcation points. To better define these points, a plot of the equilibrium points versus the parameter M_C will be useful. Finding multiple roots of an equation as complicated as the free CI rate equation is a difficult numerical task. As it turns out, a far simpler approach is to find the root(s) of Equation (2.23) as a function of M_C for given values of C. If the variation of the rate equation as a function of M_C is considered, that is:

$$\frac{d}{dM_C}\left(\frac{dC}{dt}\right) = \delta_C\left(1 + 2\frac{H_S C}{D_T \varepsilon_S^{H_S}}\left(\frac{M_T}{1+(C/\varepsilon_R)^{H_R}} - 2\frac{D_C}{D_T}(M_C - C)\right)^{H_S - 1}\right)$$

then it can be seen that the slope of the rate equation as a function of M_C will be positive



(a) Steady states of free CI concentration

(b) Steady states of *pR* activity

Figure 2.7: Equilibrium solutions for the Tum−CI MFL as a function of the steady-state concentration of total CI, which is set by parameter M_C in the steady-state analysis of the free species rate equations defined in the text. Solid lines mark the stable states of **(a)** free CI concentration, C, or **(b)** *pR* activity, whilst the dotted lines mark unstable states of those variables.

for any choice of C. That is, the rate equation is *strictly increasing* as a function of M_C, which means that there can be at most one root in terms of M_C.

The root in terms of M_C was found for a range of C concentrations using numerical root finding algorithms and optimisation routines in R, and this data then plotted, after swapping the axes, in Figure 2.7(a). This shows the variation of the steady states of the Tum−CI MFL as a function of the parameter setting the maximum level of CI. Recall that this parameter is simply defined as the CI production rate from the P_{lac} promoter divided by the CI degradation rate, and as shown in Section 2.2.4, gives the steady-state concentration of total CI, C_{tot}. Hence, this parameter stands as a proxy for changes in CI production rate, which can easily be implemented using inducible promoters such as P_{lac}. This then reveals a simple experimental mechanism for adjusting system parameters until a bistable state is found. What can be seen in Figure 2.7(a) is that the level of free CI slowly increases as CI production increases, but then abruptly splits into three states (two stable, one unstable) soon after M_C increases above 0.5 μM. Then just before M_C reaches 1.5 μM, bistability abruptly disappears, but the steady-state concentration of free CI continues to increase linearly as a function of M_C as might be expected.

The proportion of total CI that is free is not easy to measure experimentally, whether *in vitro* or *in vivo*, and the total concentration of CI shows no indication of bistability as discussed for the system of ODEs in terms of total species concentrations. A far easier variable to measure would be the total concentration of Tum, though this would also necessitate a modification of Tum to include a reporter. Instead, an *in trans pR* reporter gene construct

will be used as a proxy for total Tum concentration to measure the state of the Tum−CI MFL. Use of the *lacZ* gene as a reporter is well-precedented in quantitating promoter activity for phage 186. So shown in Figure 2.7(b) are the *pR* promoter activities expected for the steady-state concentrations of *C* shown in Figure 2.7(a). These have been calculated using the Tum production term from the system of ODEs in terms of free species (Equation (2.17b)), but normalised instead to steady state LacZ units (recall that the *pR* promoter produces 860 LacZ units of activity when unrepressed). This curve has exactly the same bifurcation points but has an inverted form.

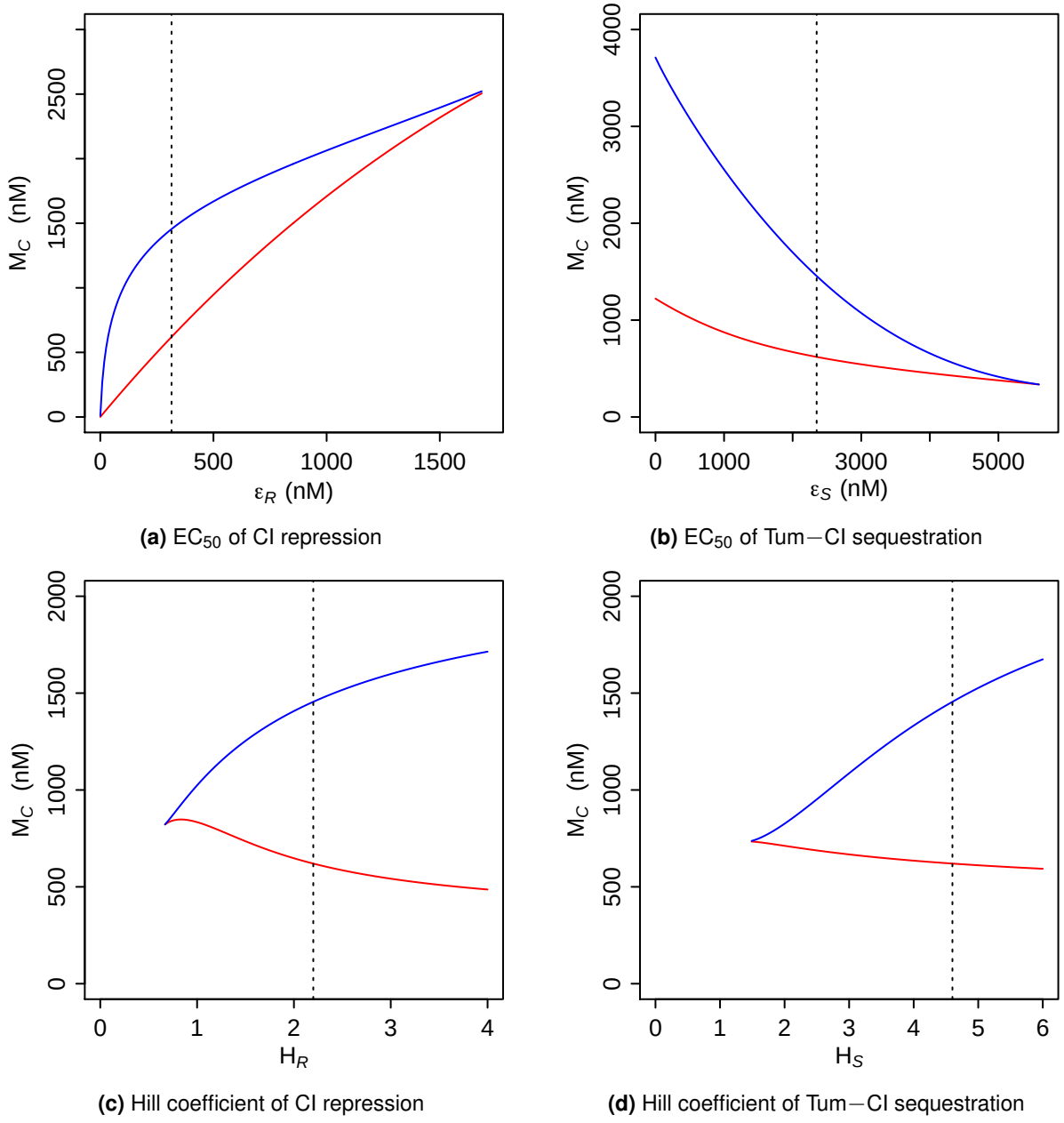
2.3.2 Varying the parameters

The points of bifurcation in terms of the maximum CI concentration, M_C, occur at the local minimum and maximum of the M_C roots of the free CI rate equation (given in Equation (2.23)) calculated as a function of free CI concentration. That is, the points of bifurcation occur at the local minimum and maximum of the graph in Figure 2.7(a) when the axes are swapped. These points, which can be found using numerical optimisation (minimisation/maximisation) routines in R, mark the boundaries of the bistable region.

Since M_C is one of the easiest parameters to adjust experimentally, it is instructive to consider how the size of this bistable region is affected by the other parameters in the steady-state formulation. These can influence where or even if the bistable region exists, thus also indicating the feasibility of synthesising a bistable Tum−CI MFL if some of the parameter estimates made in the previous section turn out to be far different than anticipated.

The parameters characterising the Tum−CI and CI−*pR* equilibria are considered first, and plots of the M_C bifurcation points versus each parameter in question are shown in Figure 2.8. The EC_{50} for CI−*pR* repression, ε_R, primarily affects the location of the bistable region, but for high enough values, the bistable region disappears so that only monostable states are accessible (Figure 2.8(a). ε_R reflects the strength of the CI−*pR* interaction, so that as ε_R increases, the strength of the interaction decreases, and higher levels of the CI production (M_C) are required to access the bistable region. Conversely, increasing the EC_{50}, ε_S, for the Tum−CI interaction leads to smaller required levels of CI production to balance the weakened strength of the Tum−CI interaction (Figure 2.8(b)). This forces the bifurcation points together until they converge and the capacity for bistability is lost.

Increasing either of the Hill coefficients, H_R or H_S, for the CI−*pR* or Tum−CI interactions leads to an increased size of bistable region (Figures 2.8(c) and 2.8(d)). This amounts to increasing the ultrasensitivity of either of those reactions; in the context of the system of ODEs in terms of total species this amounts to (asymmetrical) enhancements of the sigmoidal shape of the production rate term. In contrast, as the Hill coefficients are reduced, the size of the bistable region decreases until it disappears. In fact, bistability can still be obtained even if the CI−*pR* interaction is non-cooperative, that is, H_R is reduced below



(a) EC_{50} of CI repression

(b) EC_{50} of Tum−CI sequestration

(c) Hill coefficient of CI repression

(d) Hill coefficient of Tum−CI sequestration

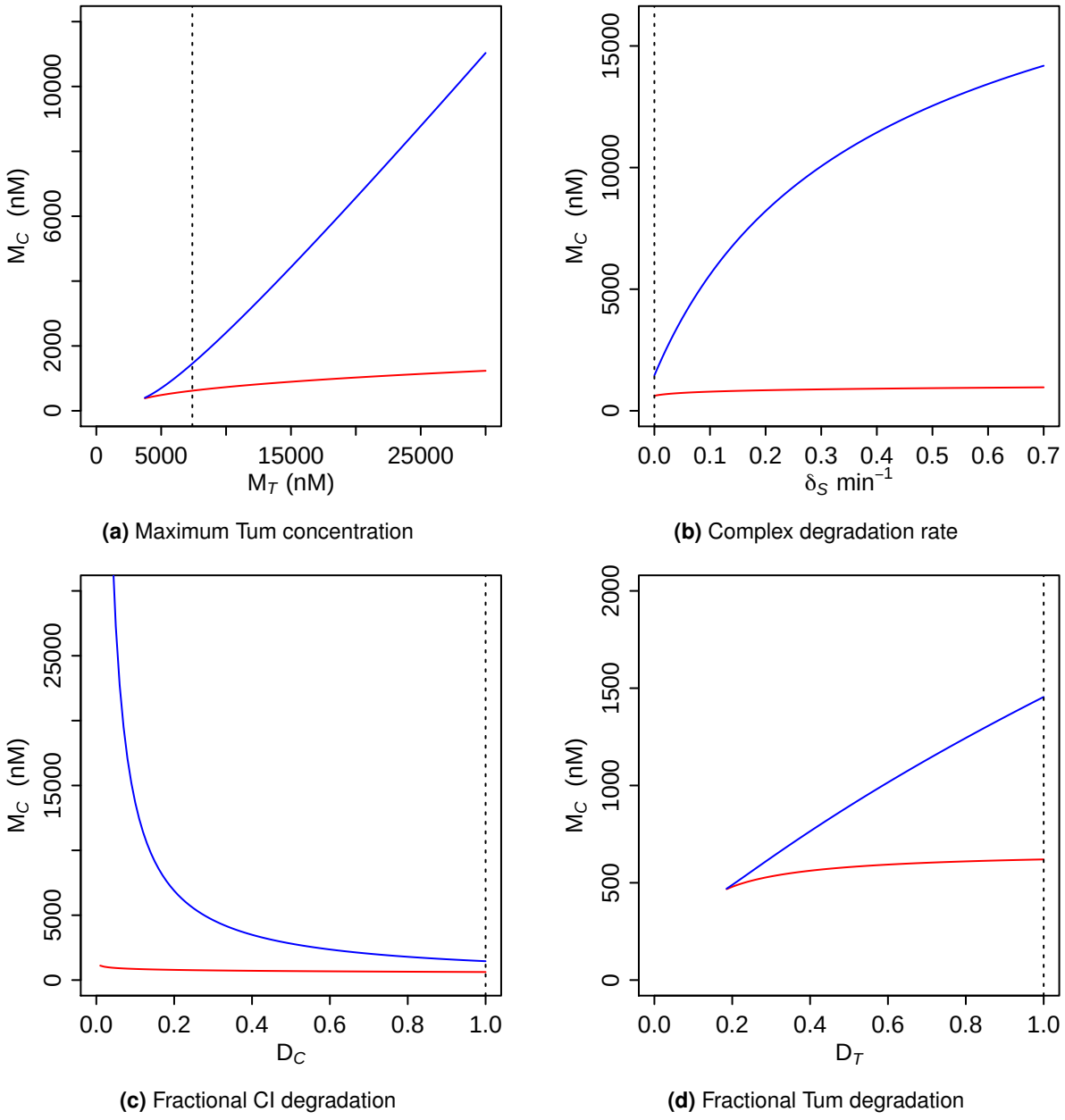
Figure 2.8: Observing variation in bistable region location as a function of parameters describing the Tum−CI and CI−*pR* equilibria. Plotted as a function of each indicated parameter are the bifurcation points of the Tum−CI MFL in terms of values of M_C(the maximum CI concentration); between these values the system shows bistability. Above the blue (upper) line and below the red (lower) line, only single monostable states exist. Initial parameters are set to those used for the steady-state curves in Figure 2.7 and are kept fixed at these values aside from the parameter in question; the vertical dotted lines indicate the position of the initial parameters before variation.

1. Indeed, François and Hakim [2005] considered *only* the case of non-cooperative interactions in their model of the MFL, but were still able to find parameter regimes supporting bistability. For the Tum−CI MFL too, the capacity for bistability can be retained even in the non-cooperative case (that is, $H_S = H_R = 1$), providing either ε_S is sufficiently decreased, or M_T sufficiently increased. The existence of bistability without cooperativity can be rationalised by the fact that sequestration of a species, such as the sequestration of CI by Tum, can produce ultrasensitivity in and of itself [Buchler and Louis, 2008].

Plots of the M_C bifurcation points versus the production and degradation rate parameters are shown in Figure 2.9. The parameters for maximum CI and Tum concentrations, M_C and M_T, are related to both production and degradation rates of those species. Either an increase in production rate or decrease in degradation rate lead to increases in those parameters. Since production rates are more easily modified experimentally it makes more sense to refer to and think of these parameters as influencing the production rates. Increasing the maximum Tum concentration parameter, M_T, simply increases the size of bistable region indefinitely (Figure 2.9(a)). This is primarily due to the increase in the production rate (M_C) of CI that is required to reach the high *C* bifurcation point.

Thus far it has been assumed that Tum does not target CI for degradation, that is, the rate of targeted degradation of the complex $\delta_S = 0$. In that case, the degradation rates of either CI or Tum scale only the magnitudes of M_C and M_T, since if $\delta_S = 0$, the fractional degradation rate adjustment factors, D_C and D_T, are always one. However, if δ_S is non-zero, then the steady-state equations are shaped by D_C and D_T, which give the fraction of degradation attributable to the free species as a proportion of the combined degradation for free and complex-bound forms. Hence at this point, for a practical consideration of targeted degradation of the complex, additional estimates on the scales of both CI and Tum degradation rates need to be made. Little has been determined in terms of the stability or otherwise of the CI and Tum proteins, though the CI repressor is believed to be stable (not targeted for degradation). Stable proteins are only lost by dilution as the cells grow in volume and divide. For fast-growing *E. coli* a typical cell doubling time is 30 minutes, which gives a rate of dilution $\gamma = \log(2)/30 \approx 0.023\ \text{min}^{-1}$. This sets a base rate of loss of each species, and any significant contribution of targeted degradation must occur at comparable or faster rates.

If both the degradation rate of CI, δ_C, and the degradation rate of Tum, δ_T, are set to the dilution rate (such that targeted degradation of either free species is zero), then variation of δ_S simply amounts to varying ε_S. This can be seen by noting that the ratio D_C/D_T in that case would always be one, whilst the D_C term appears elsewhere to scale the ε_S parameter in Equation (2.23). In other words, an increase in δ_S results in a decrease in D_C, which then scales the magnitude of ε_S. A more interesting response occurs when the half-lives differ. Protein half-lives resulting from targeted degradation may be as short as 2 minutes as seen for the phage λ CII protein [Shotland et al., 1997]. To explore this possibility, the



(a) Maximum Tum concentration

(b) Complex degradation rate

(c) Fractional CI degradation

(d) Fractional Tum degradation

Figure 2.9: Observing variation in bistable region location as a function of parameters that characterise the production and degradation rates of the Tum−CI MFL. Plotted as a function of each indicated parameter are the bifurcation points of the Tum−CI MFL in terms of the maximum CI concentration, M_C, between which points the system shows bistability. Above the blue (upper) line and below the red (lower) line, only single monostable states exist. Initial parameters are set to those used for the steady-state curves in Figure 2.7 and are kept fixed at these values aside from the parameter in question; the vertical dotted lines indicate the position of the initial parameters before variation.

half-life of Tum is set ten-fold shorter (to 3 minutes), that is, Tum degradation is set ten-fold faster than that of CI (which is kept dilution limited). With these choices of degradation rate, increasing the degradation rate of the complex from zero to a rate of degradation equivalent to a half-life of 1 min, greatly increases the M_C required to reach the upper bifurcation point (Figure 2.9(b)). This trend can be attributed mainly to the faster decrease of D_C than D_T as δ_S increases, which results from the choice of a slower rate of degradation, δ_C, for CI. In effect, degradation of the complex acts to magnify sequestration by siphoning off the complex as it is produced. If the half-lives of the free species are also unbalanced, as for this case where Tum degradation is much faster, then the slower-degrading CI is most affected, with much more of it required to outcompete the extra loss produced by complexation.

For completeness, the variation of the bifurcation points in terms of the parameters D_C and D_T are also shown (Figures 2.9(c) and 2.9(d)). Given their dependence on the degradation rate of the complex, these factors are harder to interpret physiologically. However, a decrease in D_C can be thought of as decreasing the rate of free CI degradation below that of the complex whilst keeping free Tum degradation much faster than that of the complex so that D_T remains at unity, and vice versa for a decrease in D_C.

2.4 Hysteretic behaviour

The steady-state analysis of the Tum−CI MFL has shown that with a suitable choice of parameters, the genetic network has the capacity to produce two stable states. However, up until this point it has not been considered how each of those stable states might be accessed, that is, how switching between these two stable states might be achieved experimentally. The phenomenon of hysteresis — whereby a system 'remembers' its past state as parameters are varied — is a feature expected of all bistable networks [Ferrell, 2002]. This motivates an assay for bistability that exploits the hysteresis anticipated as the production rate of CI is varied.

In Section 2.3.1, it was shown that by balancing the steady-state concentration of total CI (M_C) against the other parameters, a region of bistability flanked by two monostable regions could be found. With CI production from an inducible P_{lac} promoter, the steady-state concentration of total CI is easily changed by modifying the concentration of inducer, in this case IPTG. If the production rate is set such that the system operates in one of the monostable regions, the state of the system is drawn to that monostable state. Once at equilibrium, if the production rate is then adjusted to take the system into the *bistable* region, the system should prefer the stable state that is closest to the monostable state it started in. This concept is illustrated and explained for the Tum−CI MFL in Figure 2.10. As alluded to earlier, the state of the switch will be monitored by the activity of the *pR* promoter. This underpins the shape of the hysteresis curve as illustrated in the figure — it

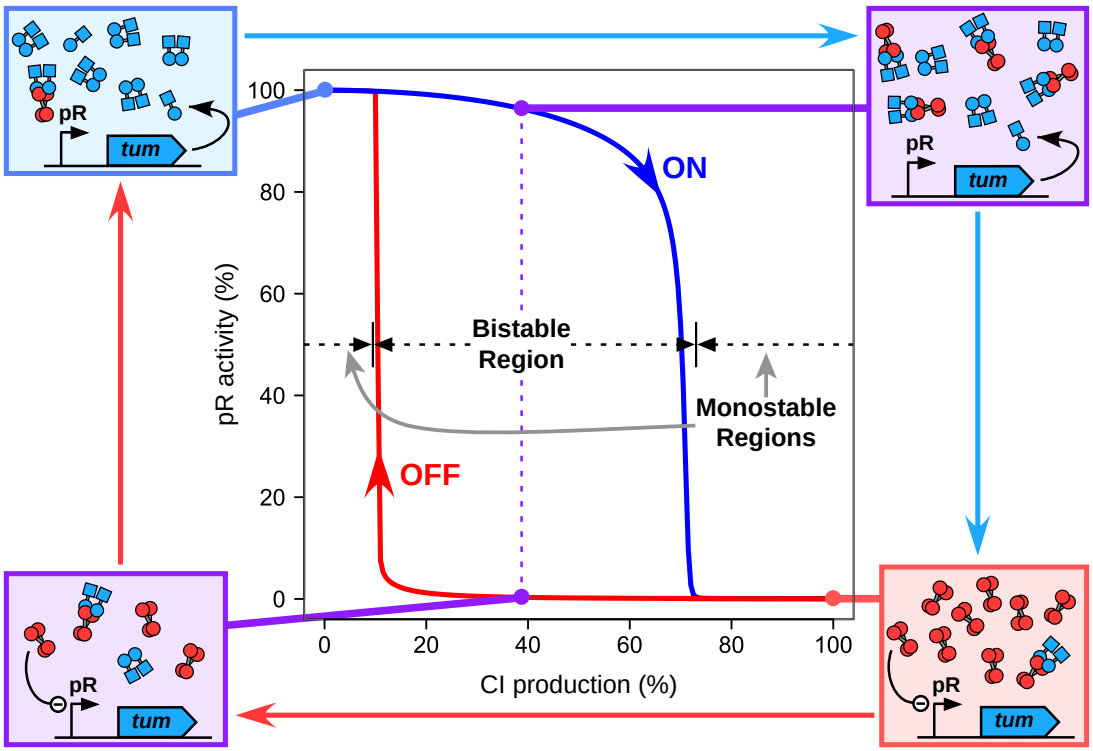


Figure 2.10: The Tum−CI MFL should exhibit hysteresis, a memory of state, which can be understood qualitatively by considering the effect of varying the CI production rate. Starting from the top left panel and proceeding clockwise, if production of CI is low enough, the MFL operates in a monostable regime where the *pR* promoter is always left unrepressed resulting in high activity and production of Tum. As CI production is increased from that level, the system follows the blue 'on' curve into the bistable region, since the effect of an increased CI concentration is absorbed by sequestration due to the pre-existing high level of Tum (top right panel). Eventually, a threshold is reached, after which point the concentration of CI overcomes the pool of Tum, and the system enters the alternative monostable regime where *pR* is always held repressed (bottom right panel). In contrast, if the production rate of CI is decreased to take the system from this monostable region into the bistable region, the 'off' curve is followed, since without sufficient Tum to relieve repression, CI continues to actively repress *pR* (bottom left panel). Again a threshold is reached when CI repression is so low that it cannot outcompete the level of expression of Tum from *pR*.

is based on the steady-state plot in terms of *pR* activity shown in Figure 2.7(b). In practical terms, the assay can be described:

1. grow cultures overnight in either high or low IPTG to set the switch in one of the monostable regions,
2. subculture into intermediate IPTG concentrations in the bistable region and grow to log phase to allow the system to equilibrate,
3. measure the production rate from *pR* by reporter gene assay.

Depending on which monostable region the system starts in, whether the low IPTG, low CI production state, or high IPTG, high CI production state, the Tum−CI MFL will follow either the 'on' or 'off' curves respectively. Cultures that end in the bistable region with the same final conditions (same IPTG), but alternative initial conditions are expected to result

in different final *pR* activities. This hysteresis would not be expected for a similar assay of a monostable system.

Assays of hysteresis have been used elsewhere to characterise bistability, including for a bistable positive autoregulatory circuit synthesised in mammallian cells [Kramer and Fussenegger, 2005], for a synthetic sequestration-based switch in *E. coli* [Chen and Arkin, 2012], for the natural *lac* operon [Ozbudak et al., 2004], and for a modified yeast galactose network [Venturelli et al., 2012]. Characterising bistability in the Tum−CI MFL using hysteresis is thus well precedented, and will be the primary assay for bistability used in this work.

2.4.1 Time course simulations of the Tum−CI MFL

To confirm that the present model of the Tum−CI MFL supports hysteresis as described, the system of ODEs in the free species concentrations (Equation (2.17)) can be solved over time, for given choices of initial conditions. A number of the necessary parameters were derived in the context of the steady-state model (see Section 2.3.1), but to further model dynamics these must be complemented by a number of additional parameters that set timescales for each process.

Degradation rate parameters for CI, Tum and the Tum−CI complex have already been discussed to a limited extent in Section 2.3.2. To start with here, both Tum and CI are assumed dilution limited, that is, with a half-life equal to a typical cell doubling time of 30 minutes so that $\delta_C = \delta_T = \gamma$. However, the alternative choice with Tum degradation set 10 times faster will also be considered; additionally accounting for dilution Tum degradation becomes $\delta_T = 10\delta_C + \gamma = 11\gamma$. These choices of degradation rate then also set the production rates, since these must balance to produce the steady-state maximum concentrations of CI and Tum, M_C and M_T. With the choices of M_C and M_T made earlier, then:

$$\mathrm{p}_C = \mathrm{M}_C\delta_C = (1500\ \mathrm{nM}) \times (0.023\ \mathrm{min}^{-1}) \approx 35\ \mathrm{nM.min}^{-1}$$
$$\mathrm{p}_T = \mathrm{M}_T\delta_T = (7400\ \mathrm{nM}) \times (0.023\ \mathrm{min}^{-1}) \approx 170\ \mathrm{nM.min}^{-1}$$

However, note that in the hysteresis assay, p_C is allowed to vary and the initial states must be chosen well within the monostable regions. For this reason, the low p_C state will be set to 0 $\mathrm{nM.min}^{-1}$ and the high p_C state to 50 $\mathrm{nM.min}^{-1}$, with final rates set between those extremes. For the case where Tum degradation (δ_T) is set 11 times faster than dilution, consistency with the steady-state model M_T is maintained, so that p_T must similarly be set 11 times faster to balance the faster degradation rate. Targeted degradation of the complex is ignored, that is, $\delta_S = 0$.

With a focus on the assay, another important consideration is the time evolution of the *in trans* reporter product of the *pR* promoter. For a *pR–lacZ* reporter, the activity of the

β-galactosidase (β-gal) enzyme product of the *lacZ* gene is measured and assumed propotional to the concentration of enzyme in the cells. As with Tum production, the production rate of *lacZ* is reduced by repression so that the evolution of β-gal, Z, is described by:

$$\frac{\mathrm{d}Z}{\mathrm{d}t} = \frac{\mathrm{p}_Z}{1 + \left(\frac{C}{\varepsilon_R}\right)^{\mathrm{H}_R}} - \delta_Z Z \tag{2.24}$$

where the maximum production rate, p_Z, and degradation rate, δ_Z, need to be determined for the β-gal enzyme. Since the enzyme activity (proxy for concentration) is reported in LacZ units, these are the units chosen to represent β-gal concentration, Z. This means that the maximum β-gal concentration from this promoter can be set using the β-gal activity for unrepressed *pR* stated above, that is, $\mathrm{M}_Z = 860$ LacZ units. β-gal is assumed stable so that its degradation rate is given by $\delta_Z = \gamma = 0.023\ \mathrm{min}^{-1}$, and hence the unrepressed production rate $\mathrm{p}_Z = \mathrm{M}_Z\delta_Z \approx 20$ LacZ units.min^{-1}.

The only parameter that remains to be determined is the off-rate for the Tum−CI sequestration reaction, which sets the timescale of the Tum−CI equilibrium reaction. Since this parameter is only a representative off-rate for a multi-step reaction scheme, an estimate is made by assuming the reaction simply follows second order kinetics with association constant $K = 1/\varepsilon_S$, the EC_{50} for the cooperative process. Then by assuming diffusion-limited dynamics for the representative two-species binding reaction, an on-rate, k_S, can be estimated using protein diffusion rates in *E. coli*, and a typical off-rate determined using the relation $\mathrm{k}_{-S} = \mathrm{k}_S/K = \mathrm{k}_S\varepsilon_S$. Diffusion of proteins in *E. coli* is dependent on shape, charge and size. However, for a small protein like Tum (approximately 40 kDa), diffusion would be close to that measured for green fluorescent protein (GFP) (approximately 60 kDa) with a rate of diffusion in *E. coli* ranging from $4.6 - 7.7\ \mu\mathrm{m}^2.\mathrm{s}^{-1}$ depending upon whether the protein is His_6-tagged Elowitz et al. [1999]. Using a diffusion coefficient, D, the steady-state flux of molecules into a spherical interaction area can be used to derive on rates for diffusion-limited reactions [Sneppen and Zocchi, 2005]. For a single reaction centre, the rate of molecules entering the reaction zone is given by $4\pi\epsilon Dc$, where ϵ is the radius of the target binding region and c is the concentration of the diffusing species.

To generalise to a reaction rate for two diffusing species, the diffusion coefficient should be doubled [Phillips et al., 2009]. Hence, choosing ϵ to be 6 nm (roughly the diameter of the amino-terminal domain (NTD) of CI) the on-rate for an *in vivo* second order reaction can be estimated at:

$$\begin{aligned}
\mathrm{k}_S &= 8\pi\epsilon D \quad (\mu\mathrm{m}^3.\mathrm{molecule}^{-1}.\mathrm{s}^{-1}) \\
&= 8\pi\epsilon D \times (10^{-15}\ \mathrm{L}.\mu\mathrm{m}^{-3}) \times (N_\mathrm{A} \times 10^9\ \mathrm{molecules}.\mathrm{nmol}^{-1}) \times (60\ \mathrm{s}.\mathrm{min}^{-1}) \\
&= 25\text{–}42\ \mathrm{nM}^{-1}.\mathrm{min}^{-1}
\end{aligned}$$

where N_A is Avogadro's constant. Using this estimate and the estimate for ε_S determined in Section 2.2.2, the timescale for the Tum−CI sequestration reaction will be around $\mathrm{k}_{-S} =$

$\varepsilon_S k_S = 5.8\text{–}9.6 \times 10^4$ min^{-1}. This means that the Tum−CI equilibrium reaction should be *much* faster than the species degradation rates, lending further credence to the total species model (Equation (2.18)) where the Tum−CI reaction was assumed always at equilibrium.

With the parameters set, all that remains is to choose the initial conditions. In the hysteresis assay, growth overnight in the monostable regions is used to take the system close to each monostable steady state. In the modelling, this can be equivalently represented by first choosing concentrations C, T, S, and Z that are close to their equilibrium values when p_C is set to either 0 or 50 nM.min^{-1} and then simulating growth in the monostable states for sufficient time to ensure that equilibrium is reached. The approximate equilibrium states are determined by first calculating the steady-state concentration of total CI, C_{tot}, that is produced in each of the monostable regions. These concentrations are then used as a first guess for free CI, C, so that an estimate of the steady-state concentration of total Tum, T_{tot}, can be made. Using these values for C_{tot} and T_{tot}, the Tum−CI equilibrium equation (Equation (2.19)) is solved for free CI and an improved guess for T_{tot} made using this new estimate. Then the concentrations of the free species can be calculated from these C_{tot} and T_{tot} estimates using the equilibrium and mass balance equations. The initial concentration of Z is simply set to the steady-state value calculated using the final estimate for free CI concentration.

The system of ODEs was solved numerically in R using the package deSolve [Soetaert et al., 2010], which provides interfaces to ODE solvers in R. Due to the vastly different timescales of the Tum−CI equilibrium reactions versus the production and degradation rates, the system of equations is expected to be stiff, requiring very small timesteps only for the balanced complex formation and degradation terms. As a result, the `lsoda` function was used, which provides an interface to the FORTRAN routine of the same name [Petzold, 1983; Hindmarsh, 1983] that automatically switches between stiff and non-stiff solvers.

To match the assay design, the system of ODEs is simulated from the initial conditions described above at the overnight P_{lac} promoter production rate (0 or 50 nM.min^{-1}) for 100 minutes to ensure equilibrium in each monostable state. Then the production rate p_C is switched to an intermediate rate and simulation proceeds for another 6 hours (300 minutes). Simulations of the transition from one monostable state to the other and from each monostable state to the same production rate within the bistable region are shown in Figure 2.11.

When starting from a CI production rate in the low monostable region, the concentration of Tum equilibrates close to its maximum value, but the absence of CI results in very little complex being present. If the production rate of CI is increased into the high monostable region, to begin with all of the CI produced from P_{lac} is immediately sequestered by the high levels of Tum to become locked up in the complex. This reduces the concentration of free Tum, and as CI levels continue to rise to their new equilibrium value, eventually the concentrations of both free Tum and the complex are reduced to zero.

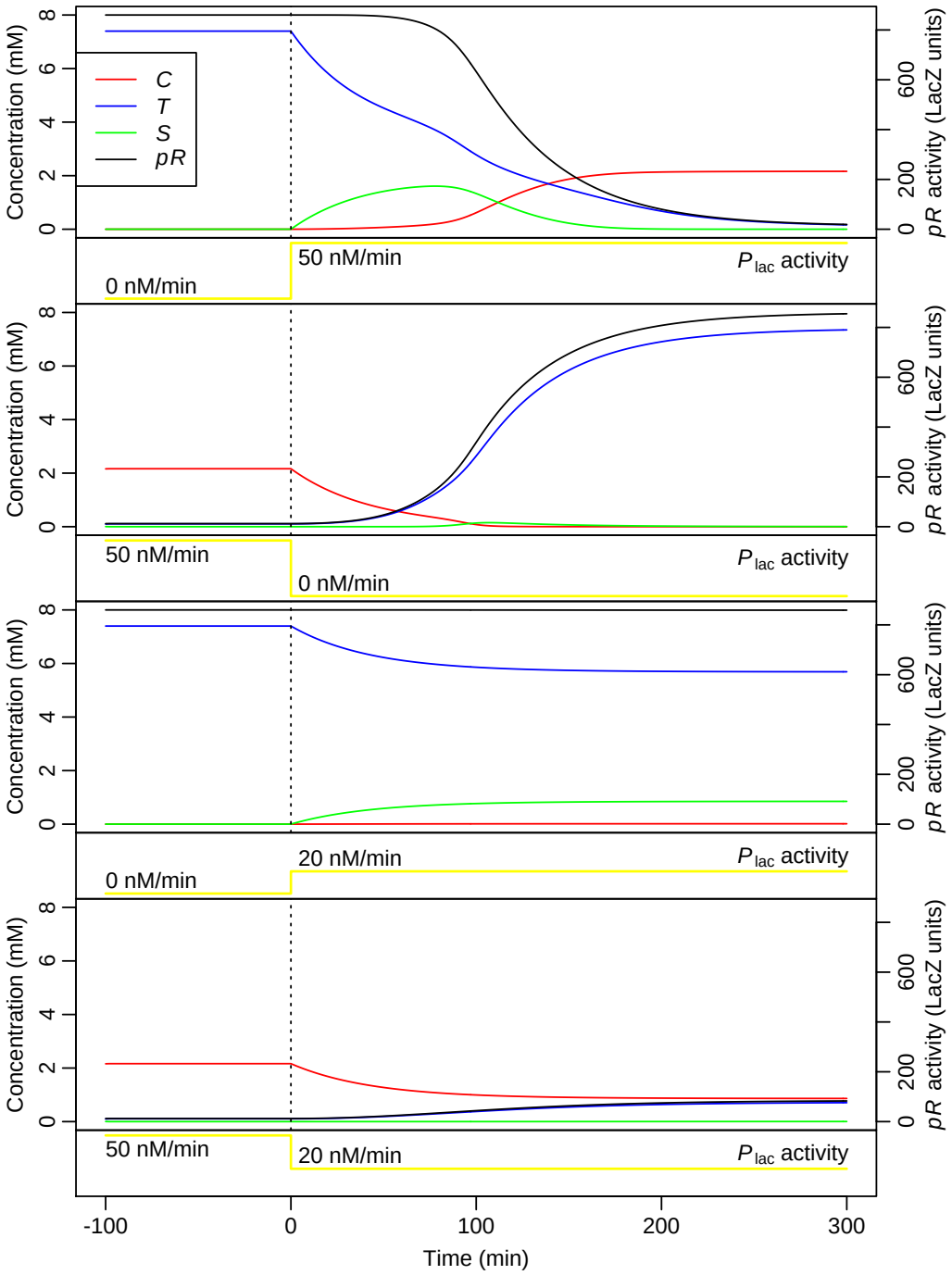


Figure 2.11: Deterministic time course simulations of the Tum−CI MFL confirm the predictions of the hysteresis assay. Shown are time course trajectories of the concentrations of free CI, C, free Tum, T, and the Tum−CI complex, S, as determined by the system of ODEs in terms of those variables (Equation (2.17)). Vertical dotted lines mark the point at which the rate of production of CI from P_{lac} is changed from its initial rate to final rate; the specific P_{lac} activities are depicted beneath each simulation. The concentration of the reporter, β-galactosidase, was also simulated to determine the *pR* activity that would be measured in the assay (black curve). This was simulated in LacZ units so has been rescaled according to the right axis. Shown from top to bottom are simulations from each monostable region to the other (low $\mathrm{p_C}$ to high $\mathrm{p_C}$ and vice versa), and simulations from each monostable region to a single point in the bistable region.

In contrast, starting in the high monostable region leaves the initial concentrations of free Tum and the complex close to zero, with the concentration of free CI at its highest value. If the production rate is decreased into the low monostable region, the concentration of free CI decreases, and relief of repression at the *pR* promoter is observed. As free Tum increases in consequence, slight acceleration of the loss of free CI is observed when a small concentration of the Tum−CI complex appears. Past this point, free Tum continues to equilibrate to its maximum value.

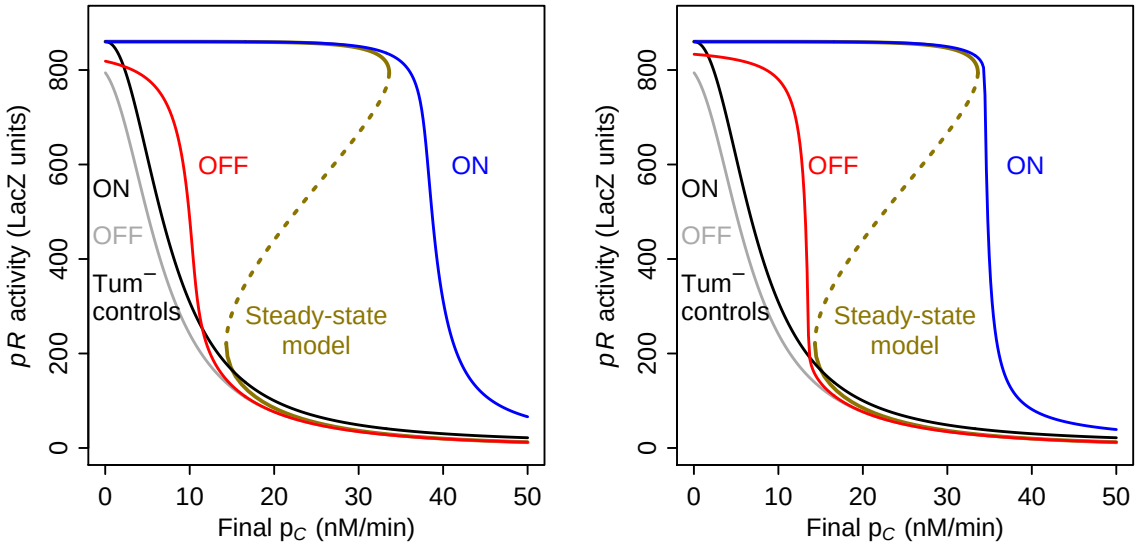
It is worth noting the time it takes for these two extreme switching cases to reach a state close to equilibrium. Growth overnight takes upwards of 12 hours (720 minutes), so this is a good guarantee that equilibrium in the monostable regions is reachable. However, after subculturing 1/500 from an overnight (stationary phase) culture, a culture of *E. coli* typically takes only around 3–4 hours to grow to log phase (an optimal point for measurement by β-gal assay). If 3.5 hours (210 minutes) is chosen as a representative value, it can be seen that the cells will get close to but not quite reach equilibrium. This will be an important consideration if the cells reach log phase faster than expected, or dilution rates of the proteins have been underestimated (recall that these rates of loss effectively set the time it takes to reach equilibrium).

Also shown in Figure 2.11 are time course simulations from each of the monostable states to an intermediate production rate within the bistable region. In these cases, hysteresis is clear: when starting in the low monostable region with high Tum production, the high Tum production state is maintained into the bistable region; when starting in the high monostable region with low Tum production, the low Tum production state is maintained. Note that the transition from a high Tum production state into the bistable region produces no change in *pR* activity, since the increased concentration of CI is siphoned into formation of the complex due to the high levels of Tum present. In contrast, the transition from a low Tum production state into the bistable region is accompanied by an increase in *pR* activity, since the reduction in CI concentration reduces the level of repression at *pR*, but not by enough to cause the switch to flip.

2.4.2 Simulating the hysteresis assay

The time course assays illustrated in Figure 2.11 can be repeated for a large range of final P_{lac} production rates to visualise the anticipated shape of the hysteresis assay curve. Two alternative time courses are obtained for each final production rate — one for cultures starting from the low monostable region, the other for cultures starting from the high monostable region — and the activity of *pR* determined from these time courses at the typical assay time of 3.5 hours gives each point of the hysteresis loop described earlier (Figure 2.10).

For the parameters described in the previous section, the hysteresis loop shown in Figure 2.12(a) is obtained. This is overlaid on top of the steady-state curve shown earlier



(a) Dilution limited Tum

(b) Active degradation of Tum

Figure 2.12: Simulating the hysteresis loop of the Tum−CI MFL by time course deterministic modelling of the free species ODEs. Two time courses are simulated, one starting from low P_{lac} activity (p_C=0 nM/min) and the other starting from high P_{lac} activity (p_C=50 nM/min), which then switch to intermediate P_{lac} activities (x-axis) for an equilibration time of 210 minutes. The *pR* activity obtained at that time is plotted against the final rate of CI production, p_C. Similar time course assays were performed for a model omitting Tum production to produce the monostable Tum$^-$ control curves. For comparison, the steady-state *pR* curve for the Tum−CI MFL is also shown, where the maximum CI production rate, M_C, has been appropriately scaled to a production rate, p_C. **(a)** Setting the loss of Tum, δ_C, as dilution limited produces a slowly equilibrating MFL. **(b)** If the Tum degradation rate is instead set 11 times faster to simulate active degradation, equilibration occurs much more quickly.

in Figure 2.7(b) for comparison. The hysteresis assay curve matches well with the steady-state predictions over the majority of the bistable region confirming that this assay is a suitable way to characterise the stable states of the Tum−CI MFL. However, with this standard time of equilibration at the new production rate (210 minutes), the trailing ends of each curve (in the opposing monostable regions) do not make it to their steady-state values. The divergence is most evident just past each of the bifurcation points.

As a control measure for equilibration and to demonstrate the result anticipated of a monostable system, it is useful to consider the case where Tum is absent from the network, thus removing the source of positive feedback. With Tum absent, no complex formation is possible and the system of ODEs in terms of the free species reduces to the following two differential equations:

$$\frac{\text{d}C}{\text{d}t} = \text{p}_C - \delta_C C \tag{2.25a}$$

$$\frac{\text{d}Z}{\text{d}t} = \frac{\text{p}_Z}{1 + \left(\frac{C}{\varepsilon_R}\right)^{\text{H}_R}} - \delta_Z Z, \tag{2.25b}$$

which involves no feedback and hence can only exhibit monostability. The steady-state free CI concentration is simply given by $\text{M}_C = \text{p}_C/\delta_C$ and the steady-state reporter activity

by $M_Z/(1+(M_C/\varepsilon_R)^{H_R})$ — a standard CI$-pR$ repression curve. These equations can be similarly simulated over time using the same hysteresis assay protocol to produce the control curves shown in Figure 2.12(a). These curves do not overlap each other, but this is not an indication of two alternative states, but simply that equilibrium is not yet reached by this time point. They also reveal the similarity of the low pR activity stable state of the Tum$-$CI MFL with the shape of the standard CI$-pR$ repression curve.

The time to reach equilibrium is affected primarily by the slow time evolution of the degradation and production reactions. Like for the case of complex formation and dissociation discussed in Section 2.2.3, the timescale of these reactions is set by the rate of degradation. The effect of an eleven-fold faster Tum degradation rate on the shape of the hysteresis curve was also simulated and is shown in Figure 2.12(b). This change has no effect on the shape of the control curves or the steady-state curve, but does produce a hysteresis curve that is much closer to steady state when measured at the same assay time.

Nonetheless, the hysteresis curves still diverge from equilibrium near the points of bifurcation, so a relevant consideration is the time needed to produce overlapping curves. Slightly increasing the equilibration time or choosing a faster CI degradation rate would help to close the ends and the Tum$^-$ controls. More worrying is that the closer each curve gets to its bifurcation threshold, the longer the system takes to equilibrate. This is exemplified in the time course simulations in Figure 2.13 which show the progression from the low monostable region to production rates that are all within the high monostable region. The bifurcation point between the bistable and high monostable regions occurs at approximately 33.62 nM.min^{-1}. Thus in reasonable time, the hysteresis curve can only ever be expected to approach equilibrium.

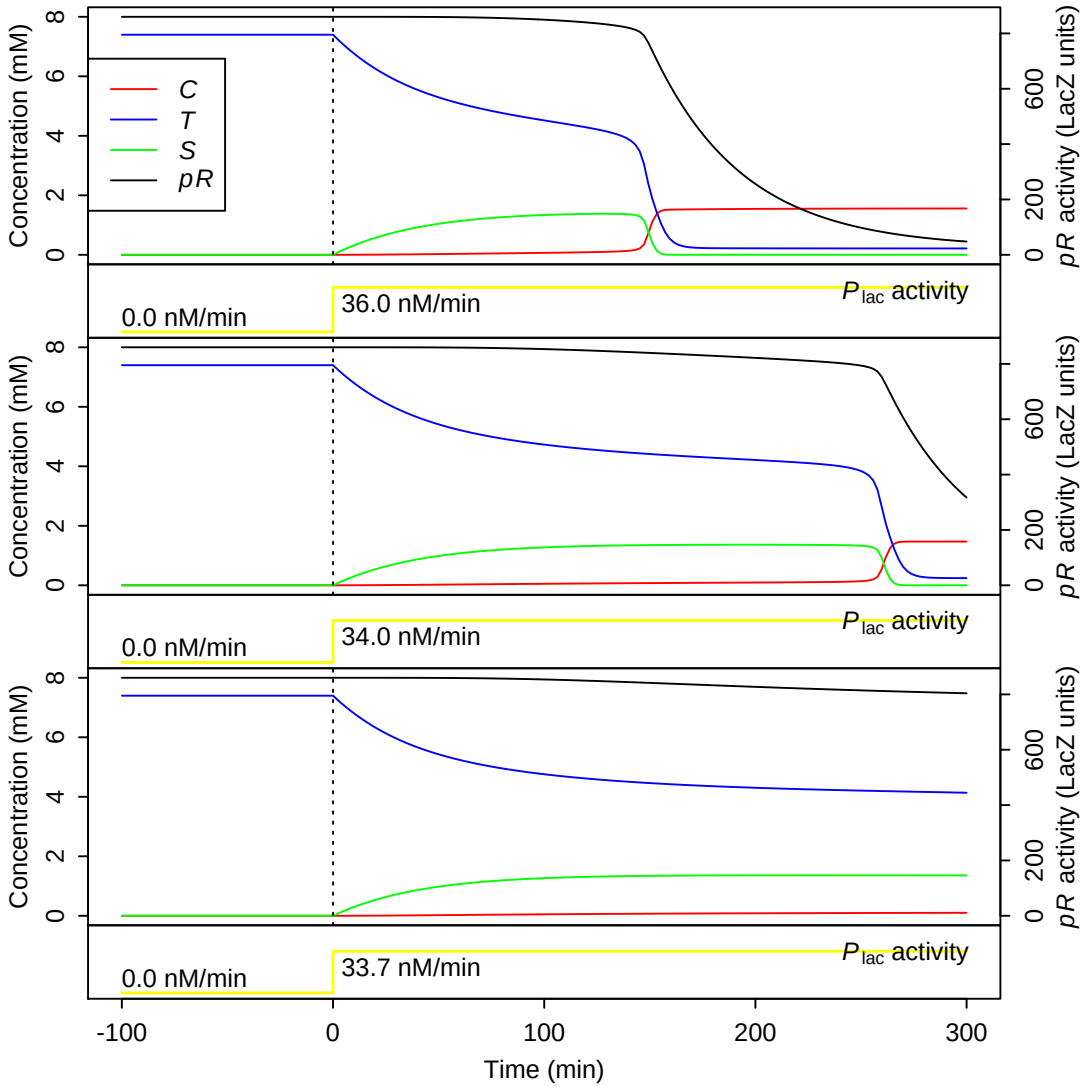


Figure 2.13: As the assayed production rate of CI nears the point of bifurcation, the time to reach equilibrium lengthens. Shown are time course trajectories performed exactly as described in Figure 2.11.

3

Designing and characterising a bistable MFL

The model of the Tum−CI MFL developed in the previous chapter showed that by placing expression of the *tum* antirepressor gene under the control of the CI-repressible *pR* promoter, it is possible to synthesize a MFL that is capable of bistability over a broad range of parameter values. Whilst derived based on sound biochemical principles, the mathematical model is nonetheless idealised, and the experimental behaviour of such a synthetic gene network is likely to deviate from its model when implemented in the complex and noisy intracellular environment [Sprinzak and Elowitz, 2005]. By basing the design on components from a bacteriophage, an independently functioning and predictable circuit is favoured, but far from guaranteed; the synthesis of fully autonomous circuits continues to remain an elusive possibility [Nandagopal and Elowitz, 2011]. In the MFL, the dependence of each component on the host environment is manifest in the reliance on host production machinery and possibly also on unrealised interactions with host proteases, since such degradation of phage proteins has been observed elsewhere [Shotland et al., 1997]. Even the formation of the Tum−CI complex may be subject to unidentified *in vivo* interactions. With these complications in mind, a good experimental design for the Tum−CI MFL will need to be both modular and flexible to compensate for the likelihood of unpredicted behaviours.

To this end, the Tum−CI MFL will be constructed from three modular systems, as illustrated in Figure 3.1, (1) the CI induction module that provides an IPTG-inducible level of CI, (2) the MFL construct module that supplies Tum from the *pR* promoter, and (3) the reporter modules that report on the level of *pR* repression. By cloning each of these systems as independent gene cassettes, the design is kept flexible and allows for easier rebalancing of each component independently of the others. As alluded to in Chapter 2, the inducible level of CI provides a way to balance the MFL and find the regions of bistability within the experimental assay. In this way, the circuit design already has in-built flexibility. Dual *gfp* and *lacZ* reporter constructs have been chosen to increase the available options for assaying *pR* activity, with the *pR-lacZ* cassette additionally chosen to facilitate comparisons with previous assays of *pR* activity [Dodd and Egan, 2002].

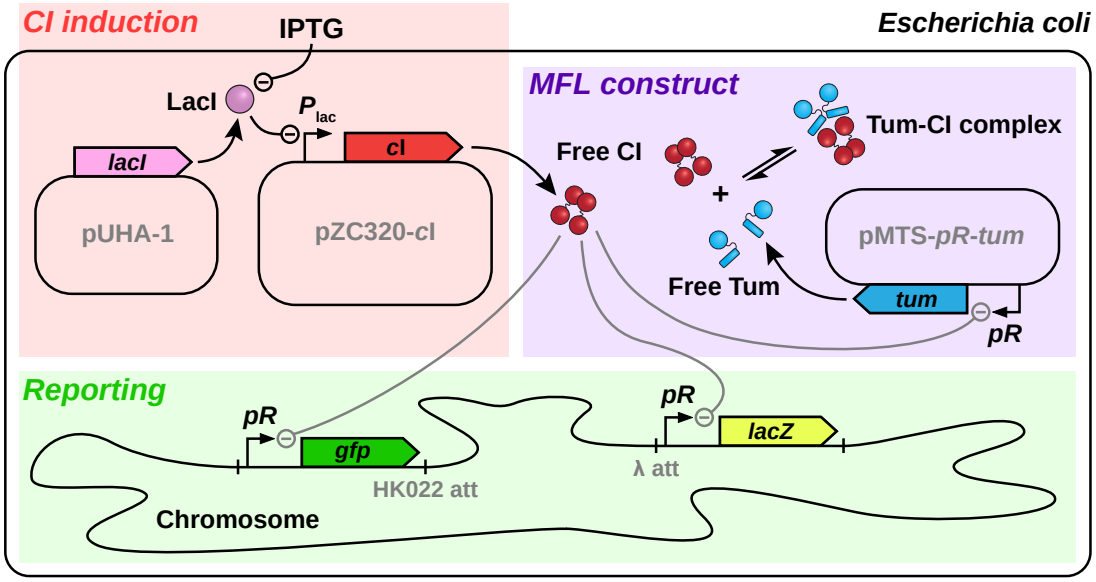


Figure 3.1: Overall strain design for the bistable Tum−CI MFL. The components can be split into three categories: (1) the MFL construct itself, with antirepressor *tum* expression from the CI-repressible promoter *pR*, located on the pMTS-*pR-tum*$^+$ plasmid, (2) the CI induction system, with *cI* supplied from the P_{lac} promoter on the pZC320-WR-*cI* plasmid and *lac* repressor (LacI) supplied from pUHA-1 to provide IPTG-inducible expression of CI, and (3) the reporter constructs, which report on the level of repression of the *pR* promoter. Dual reporter constructs mean that the activity of *pR* can be measured either via an assay of the the activity of the β-gal enzyme expressed from *pR–lacZ*, or alternatively via the fluorescence of GFP expressed from *pR–gfp*. The reporter constructs are integrated in the chromosome at the phage attachment (att) sites indicated.

This chapter begins with a presentation of the technical details and considerations required for implementing and cloning the Tum−CI MFL in *E. coli*. This is followed with an experimental description of the hysteretic assay of CI production rate, the primary assay used to test for bistability of the MFL. Formative results are presented, but these then prompt a deeper investigation of each of the components, and the remainder of the chapter is devoted to measurement and calibration of the modules. This process is necessarily quantitative to complement the modelling and assist in refining both strain and assay designs. Indeed, even a *quantitative* difference in gain through a feedback loop has the potential to produce a *qualitative* difference in output response [Wall et al., 2004], making a quantitative analysis highly relevant here. A quantitative analysis adds power to the mathematical model for testing the model's assumptions and concepts [Hasty et al., 2002], and hence this analysis further satisfies a major goal of synthetic biology to increase an understanding of design principles [Sprinzak and Elowitz, 2005; Yeh and Lim, 2007].

The quantitative characterisation of system components in this chapter is guided by the deterministic model developed in Chapter 2, and is targeted towards refining and constraining relevant model parameters. The hysteretic parameter in the model is the production rate of CI from the P_{lac} promoter, which is known to have an ultrasensitive dependence on the IPTG inducer [Palmer et al., 2009]; this dependence is characterised

here for the MFL strains. Cell doubling times and volumes are considered next to lay the groundwork for characterising rates of loss and production. Finally, experimental measurements of the rates of loss of CI, Tum and the Tum−CI complex and the rates of Tum production from *pR* and CI production from P_{lac} are presented.

3.1 Designing and cloning the MFL strains

The process of cloning a synthetic bistable Tum−CI MFL has been an iterative one, with amendments to the genetic constructs made in response to deficiencies observed during the course of assaying each construct. Much of the early strain preparation was carried out by another student, Michael Pocock, who cloned some of the primary MFL constructs [Pocock, 2007]. This section starts by summarising the work done prior to the present thesis in preparing the MFL strains, including a brief discussion of some of the hurdles that were overcome. Motivated by the results of that work, the modifications made to this preliminary system as part of this thesis are then described. These included the addition of a fluorescent reporter gene and increasing the range of CI production rates accessible by the P_{lac}-*c*I induction system.

3.1.1 Development of the preliminary Tum−CI MFL strains

The cloning and results described in this section are presented in more detail elsewhere [Pocock, 2007], but the brief summary here should be sufficient for understanding the results of the present work. Being an exercise in synthetic biology, the Tum−CI MFL was initially designed from a collection of well-studied components and genetic elements of both phage 186 and its host *E. coli* [Brumby et al., 1996; Shearwin et al., 1998; Dodd and Egan, 2002; Pinkett et al., 2006; Dodd et al., 2007b]. The strain chosen to host the MFL was NK7049 [Simons et al., 1987], listed as E4300 in this thesis. E4300 is a derivative of the widely used K-12 strain, and is useful as a reporter strain due to removal of the *lac* operon, in particular, the *lacI* and *lacZ* genes. The history of E4300 is somewhat obscure, but it has a reported genotype of Δ*lac*χ74 *galOP*308 *rpsL su*$^-$ [Simons et al., 1987; Maurer et al., 1980]. Removal of the *lac* operon (Δ*lac*χ74) is a complete deletion, but also removes additional sequences flanking the operon. Our group has observed that, when grown on L agar plates (L-plates), E4300 produces a bimodal population of small and large colonies. There is no evidence that this has an effect on assays of β-gal activity [Ian Dodd, personal communication], however, it is unknown what other effects this phenotype has on cell morphology. E4300 also grows very slowly in minimal media without added amino acids (presented in more detail later in Section 3.4.2).

To provide IPTG-inducible expression of 186 CI, the MFL design borrowed the induction system of a prior study of CI repression of *pR* in E4300 [Dodd and Egan, 2002]. Since CI is the maintenance repressor of phage 186, keeping the *pR* promoter repressed dur-

ing lysogeny when the phage is integrated in the chromosome, it is produced in single copy and operates against a single-copy promoter. The repression assays of Dodd and Egan [2002] were designed to be capable of replicating such expression levels. Hence, for induction of CI, the pZC320-WR-*c*I plasmid (labelled pZC320-186*c*I in Dodd and Egan) was used; the pZC320 backbone is based on a trimmed down version of the F plasmid, which maintains close-to chromosomal copy-numbers [Shi and Biek, 1995]. In the induction plasmid, the wild-type *c*I gene with its RBS are cloned downstream of the wild-type P_{lac} promoter, and the plasmid can be selected for via its ampicillin resistance gene. Induction of P_{lac} by IPTG occurs via inhibition of the *lac* repressor, which was supplied from the pUHA-1 plasmid with p15a origin and kanamycin resistance. For the MFL, these two plasmids were introduced as is.

Repression of *pR* was monitored by an integrated *pR-lacZ* reporter in Dodd and Egan [2002]. Turning the repressor system into an MFL could be as simple as replacing the *lacZ* gene with a *tum* gene, providing a single-copy *pR-tum* module is suitably balanced against the range of production rates from pZC320-WR-*c*I. Arguably, this should be the case, since upon SOS induction of the (chromosomally integrated) phage 186 lysogen, expression of the single-copy *tum* gene from the phage-borne p_{95} promoter acts against a lysogenic level of CI. This level of CI has been quantified and approximately corresponds to the steady-state level of CI produced from pZC320-WR-*c*I when induced at an intermediate value of 40 μM IPTG [Dodd and Egan, 2002]. Given that the *pR* promoter is moderately strong (thought to be greater than or at least comparable with the strength of p_{95}), this would suggest that even in single copy it should produce Tum in sufficient quantities to overcome at least the lowest induction levels of CI produced from the close-to single-copy pZC320-WR-*c*I.

With an integrated *pR-tum* construct in mind, and further seeking a reporter that is integrated as in Dodd and Egan [2002], it remained to adapt the repressor system to a new system with two genes expressed from *pR* at chromosomal copy-numbers. One possibility could have been a construct designed to express both the *tum* and *lacZ* genes from the same *pR* transcript. However, given that the Tum−CI interaction had not been well-characterised *in vivo* and also that the strength of the p_{95} promoter was largely unknown, it was foreseen that some rebalancing may be required. Hence, a modular design was preferred with the reporter construct cloned *in trans* of the *pR-tum* construct. In that case, the *pR-lacZ* module would act as a proxy reporter for the level of repression of *pR* in the *pR-tum* module.

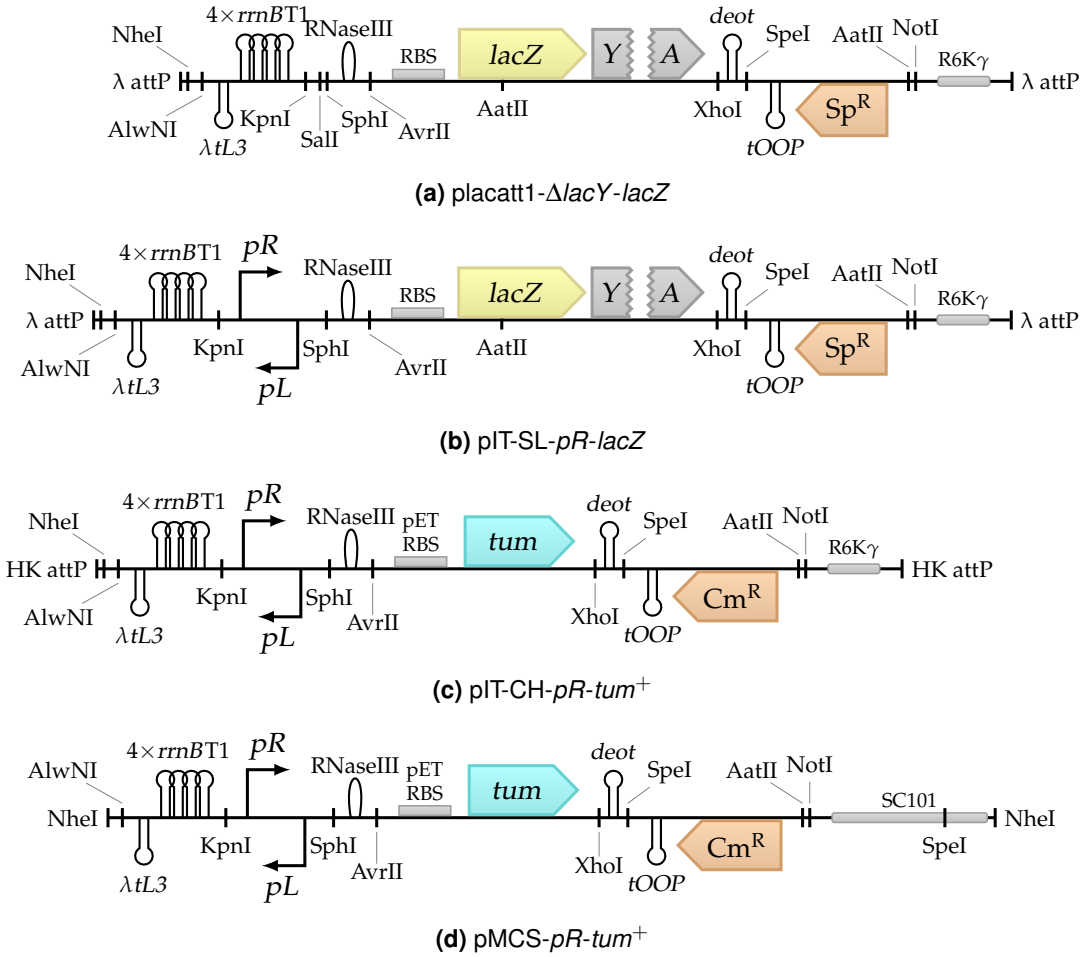
As a result, maintaining consistency between the constructs became an important consideration. Having more than one such integrated module was beyond the capacity of the system employed by Dodd and Egan [2002], where *pR* activity was assayed using constructs recombined into the λRS45Δ*YA* phage and lysogenised in single copy Dodd et al. [2001]. However, the conditional replication, integration and modular (CRIM) sys-

tem developed by Haldimann and Wanner [2001] takes advantage of phage integration mechanics to allow for integration of plasmid constructs at up to five alternative attachment (att) sites. The CRIM plasmids contain a copy of the phage att site (attP), and using helper plasmids that contain the corresponding phage integrase gene, the whole plasmid can be integrated in the host chromosome at the bacterial att site (attB). The plasmids have a modular design using unique restriction sites to simplify the task of switching the particular attP site or the selection markers (antibiotic resistances), and thus were well-suited for use in the MFL. For recombination work, the plasmids can be maintained in strains that are pir^+ since they contain the R6Kγ origin that is dependent on the Π replication factor. By integrating the modules into pir^- strains, any unintegrated plasmids will be lost.

Prior to the MFL project, an integratable *lacZ* reporter chassis had been constructed similar to those used in Dodd and Egan [2002] and Neufing et al. [2001], but based instead on the CRIM system [Ian Dodd, unpublished data]. A map of the important genetic elements of the cloning plasmid, placatt1-Δ*lacY-lacZ*, is shown in Figure 3.2(a). A particular feature of placatt1-Δ*lacY-lacZ* is the inclusion of additional terminators surrounding the expression cassette. These help to increase consistency in expression levels between the constructs, by reducing impinging transcription resulting from integration in different chromosomal contexts. Furthermore, placatt1-Δ*lacY-lacZ* inherits the ribonuclease III (RNaseIII) site from pTL61T, which helps to minimise context-dependent effects on mRNA stability [Linn and St Pierre, 1990]. Finally, placatt1-Δ*lacY-lacZ* has a deletion across the *lacY* and *lacA* genes so that expression of the permease (LacY), in particular, does not feedback on entry of IPTG into the cell.

The MFL constructs were made starting from the placatt1-Δ*lacY-lacZ* plasmid, by first cloning the *pR* promoter upstream of the reporter gene (*lacZ*) and then replacing *lacZ* with the *tum* gene. The short *pR*–*pL* fragment from the pBC2-MM-*pR*-*pL* plasmid was chosen as the template for the promoter sequence, and is the same fragment as in the equivalent pMRR9R-MM-*pR*-*pL* plasmid that is described in Dodd and Egan [2002]. It is a minimal wild-type phage 186 sequence that includes the opposing *pR* and *pL* promoters and the (mainly intervening) CI binding sites that effect repression at those promoters. Since the promoters oppose each other, transcriptional interference complicates their expression patterns [Callen et al., 2004; Sneppen et al., 2005], but the *pR* promoter is much stronger than *pL* so that transcriptional interference on *pR* due to transcription from *pL* is low [Sneppen et al., 2005]. The short *pR*–*pL* sequence does not include the flanking CI binding sites that enhance repression at *pR* by DNA looping [Dodd and Egan, 2002]. These flanking sites were excluded, since the *pR* promoter is still well-repressed without such flanking sites, and accounting for the flanking sites would greatly complicate the modelling [Dodd et al., 2007b] and consequently the synthetic design process.

As illustrated in Figure 3.2(b), the KpnI-*pR*–*pL*-SphI fragment from pBC2-MM-*pR*-*pL* was cloned into the multiple cloning site of placatt1-Δ*lacY-lacZ* to produce



(a) placatt1-Δ*lacY*-*lacZ*

(b) pIT-SL-*pR*-*lacZ*

(c) pIT-CH-*pR*-*tum*$^{+}$

(d) pMCS-*pR*-*tum*$^{+}$

Figure 3.2: Sequence maps for the *pR*-*lacZ* and *pR*-*tum* MFL modules. The maps illustrate sequence features (not to scale) for each of the plasmids as indicated, where the CRIM-based plasmids, (a)–(c), are shown linearised at the respective phage attachment (attP) sites as they would appear when integrated in the host chromosome, and the medium-copy plasmid, (d), is shown linearised at the NheI restriction enzyme site; a low-copy variant of this plasmid with the SC101* origin was also prepared. Genes are indicated by directed boxes; gene fragments have jagged ends. The *lacZ* gene is from the wild-type *lac* operon; *Y* and *A* are *lacY* and *lacA*, also from that operon, but a deletion results in non-functional products from those two genes. The *tum* gene has a His_6 tag and contains silent mutations that prevent translation of internal open reading frames 4 and 5. $Sp^{R}$and $Cm^{R}$are spectinomycin and chloramphenicol resistance genes respectively. Plasmids have either the *pir*-dependent origin R6Kγ or the medium-copy SC101 origin. Promoters are indicated by bent arrows, terminators by stem loops, ribonuclease III (RNaseIII) sites by plain loops as labelled, and ribosome binding sites (RBSs) by offset boxes (RBS: wild-type RBS; pET RBS: strong RBS obtained from the pET plasmids). Restriction enzyme sites are indicated by labelled vertical lines. **(a)** β-gal reporter chassis and precursor to **(b)** the integrable *pR*-*lacZ* reporter module, which further served as precursor to **(c)** the integrable version MFL module; **(d)** the medium-copy version of the MFL module derived from (c).

pIT-SL-*pR-lacZ* which could be integrated into the *E. coli* chromosome at the phage λ attachment site. Strong promoters like *pR* can prove difficult to clone, especially upstream of large reporter genes such as *lacZ*. For this reason, almost all DNA recombination work involving the *pR* promoter was performed in strains containing the pZC320-WR-*c*I plasmid (without pUHA-1), which provides unrepressed production of CI from the P_{lac} promoter.

The *pR-tum* module was created from the placatt1-Δ*lacY-lacZ* plasmid, with the *pR–pL* fragment introduced only in the last step, again, to avoid the complications of cloning with a strong promoter. First the *lacZ* gene in placatt1-Δ*lacY-lacZ* was replaced with the *tum*$4^{-}5^{-}$*his* gene from pET-TumHis$_6$ [Shearwin et al., 1998], via the AvrII and XhoI restriction sites. This C-terminally His$_6$-tagged variant of Tum was chosen since it had been purified and characterised *in vitro*, and provided the most quantitative and definitive data on the Tum−CI interaction at the time, making it most amenable to modelling. The *tum* gene is unusual in that it has a number of active internal open reading frames (ORFs) [Brumby et al., 1996], and a *tum* gene variant with silent mutations to inactivate the fourth and fifth ORFs simplified analysis of the purified protein. However, if anything, ORFs 4 and 5 appear to inhibit the ability of the Tum protein to induce the phage [Brumby et al., 1996], so a construct with their removal was also preferred.

For compatability with the existing integrated *pR-lacZ* reporter, both the antibiotic resistance gene and attachment site needed changing. The spectinomycin resistance gene was swapped for one providing chloramphenicol resistance using the SpeI and AatII sites. Then the attP site was exchanged from the λ attP to that of bacteriophage HK022 (phage HK) via the NotI and AlwNI sites (refer to Figure 3.2(c)). As explained above, the *pR* promoter was cloned in last, and used the same *pR–pL* fragment inserted between the KpnI and SphI restriction sites as for the *lacZ* reporter. Again, this cloning step was performed in a *pir*$^{+}$ strain containing the pZC320-WR-*c*I plasmid to keep *pR* repressed.

With the *pR-tum* and *pR-lacZ* modules integrated at the λ attB and HK022 attB (HK attB) sites respectively in E4300, the complete MFL strain became E4300 (pIT-SL-*pR-lacZ*)$_{\lambda}$ (pIT-CH-*pR-tum*$^{+}$)$_{\mathrm{HK}}$ pUHA-1 pZC320-WR-*c*I. Assays of this strain showed no deviation from the curve anticipated for repression of *pR* by CI alone [Pocock, 2007]; the presence of the *pR-tum* cassette appeared to have no effect. This implied that the Tum−CI interaction was not strong enough to relieve repression at *pR*. As already shown in Chapter 2, an effective way to increase the size of the bistable region and the efficacy of Tum is to increase the rate of Tum production.

Having been derived from the high-expression pET plasmids, the *tum* gene already had a strong RBS. So the easiest way to increase Tum production was to shift the *pR-tum* module to a multicopy plasmid. This was done by replacing the region between the NotI and NheI restriction sites, which contains the R6Kγ origin and HK022 attP (HK attP) site, with either the SC101 or SC101* origins derived from the pSC101 plasmid [Xia et al., 1991],

or the ColE1 origin, which have copy numbers of 3–4, 10–12 or 50–70 respectively [Lutz and Bujard, 1997]. The resulting chloramphenicol-resistant plasmids (the variant with SC101 is depicted in Figure 3.2(d)) were transformed into MFL strains as described above, but without the integrated (pIT-CH-*pR-tum*$^{+}$)$_{\mathrm{HK}}$ cassette. Hysteresis assays for these strains were also attempted, although strains containing the high-copy *pR-tum* plasmid with the ColE1 origin grew too slowly to be reliable in a LacZ assay. However, when the two other variants (with SC101 and SC101*) were assayed, they did display a limited degree of hysteresis. It was from this point that I took over work on the Tum−CI MFL.

3.1.2 Introducing a fluorescence-based reporter module

The hysteresis curves obtained from the early strains as described in the previous section, were ill-defined and further showed unexpected discrepancies with the modelling predictions that required further investigation. Though a number of explanations for the observed discrepancies could have been posited, experimentally distinguishing those would have been difficult in the MFL strains as detailed so far. Far more information could be extracted, however, by adding a fluorescent reporter to expand the available assay options. Many of the synthetic gene circuits studied in *E. coli* have employed fluorescent reporters for quantitation by flow cytometry [Gardner et al., 2000; Isaacs et al., 2003], or microscopy [Elowitz and Leibler, 2000; Becskei and Serrano, 2000; Stricker et al., 2008], though assay by β-gal has still been used in some cases [Atkinson et al., 2003]. It has also become an essential tool in studying genetic circuits in the context of the *in vivo* intracellular environment, since such circuits are well known to exhibit a substantial degree of stochasticity, or 'noisiness' [Elowitz et al., 2002; Cai et al., 2006]. Two major advantages of fluorescent proteins as reporters are the ability to easily make *in vivo* measurements in growing cells so that time-course measurements can be made, and, more importantly here, enabling single-cell measurements so that cell-to-cell heterogeneity can be resolved.

In contrast, a typical LacZ assay quantifies promoter activities as a population average, losing information about cell to cell variation[1]. However, this does not make the LacZ assay redundant: LacZ assay results in this system are comparable to a large body of existing data and hence more easily complement the modelling. It is also easier to obtain more sensitive measurements of promoter activity by LacZ assay. So whilst single-molecule sensitivity is indeed possible with fluorescent protein-based reporters by using total internal reflection fluorescence microscopy [Yu et al., 2006], this would require specialist equipment and significant modifications of the standard assay protocols, including a shift to minimal media, which would mean further deviating from current models of bacteriophage 186 that are optimised for protocols that have been performed in Lennox Luria broth (LB) [Dodd and Egan, 2002; Dodd et al., 2007b]. For these reasons, and given the

[1] With the aid of microfluidics and a fluorescence-based assay of β-gal activity, single-cell measurements with single-molecule sensitivity are still possible, albeit technically more challenging [Cai et al., 2006].

modular flexibility afforded by the CRIM system, the *pR-lacZ* reporter was retained and a 'dual reporter' strain cloned, which included an additional fluorescence reporter cassette.

Numerous different fluorescent reporter proteins are available with excitation and emission spectra that cover different regions of the spectrum [Shaner et al., 2005]. Choosing a fluorescent reporter that maximises measurement sensitivity requires both choosing operating wavelengths that minimise background fluorescence from the growth medium and cell constituents, and matching the spectra of each given fluorescent protein to the excitation wavelengths and emission filters available for the target instrument. In this instance, the target instrument was a flow cytometer fitted with argon and helium-neon lasers that excite at wavelengths of 488 nm or 633 nm. The enhanced green fluorescent protein (GFP) includes mutations from the wild-type GFP that move the excitation maximum closer to 488 nm and increase brightness [Patterson et al., 1997]. Another set of mutations of wild-type GFP, the cycle 3 mutations, produce better folding of the protein in *E. coli*, keeping it in the soluble fraction, since the chromophore does not become activated in the insoluble fraction [Crameri et al., 1996]. The fluorescent protein chosen for the MFL was 'folding reporter GFP', which combines the cycle 3 mutations and enhanced GFP mutations [Waldo et al., 1999], but was designed to be a fusion protein to report on protein folding of the fusion.

To maintain consistency with the *pR-tum* module, the *pR-gfp* cassette was cloned by replacing the *tum* gene of pIT-CH-*pR-tum*$^+$ with folding reporter GFP between the NdeI and BsiWI sites (see Figures 3.3(a) and 3.3(b)). Since the folding reporter GFP template was designed for use as a fusion protein, an initiation codon and valine codon were prepended as part of the cloning, so that the gene would be independently translated and match the beginning of wild-type GFP (see Materials and Methods (Chapter 8) for specific cloning details). As done for the single-copy *pR-tum* version of the MFL, this construct was integrated in single-copy at the phage HK attachment site, keeping the *pR-lacZ* reporter in the phage λ attachment site. The base dual reporter strain is E4300 (pIT-SL-*pR-lacZ*)$_\lambda$ (pIT-CH-*pR-gfp*)$_{\mathrm{HK}}$ pUHA-1, but will be referred to as E4300DR for brevity. Note that since the pUHA-1 plasmid (supplying *lac* repressor) is a component of all MFL strains using this induction system, the dual reporters were integrated into E4300 pUHA-1, so that the dual reporter strain (E4300DR) also contains the pUHA-1 plasmid.

Even with just a single copy of the chloramphenicol selection marker gene (designated Cm$^{\mathrm{R}}$in the figures), the dual reporter strains could compromise maintenance of the chloramphenicol-resistant pMCS-*pR-tum*$^+$ plasmid. So in addition to creating a new reporter construct, the two *pR-tum* plasmids (that is, both medium- and low-copy variants) were revised by replacing the Cm$^{\mathrm{R}}$gene in pMCS-*pR-tum*$^+$ (see Figure 3.3(c)) with one providing tetracycline resistance (Tc$^{\mathrm{R}}$) between the SpeI and AatII sites to produce pMTS-*pR-tum*$^+$ (see Figure 3.3(d)). The presence of an additional SpeI site in both the SC101 and SC101* origins meant that three-fragment ligations were necessary in both cases, and the

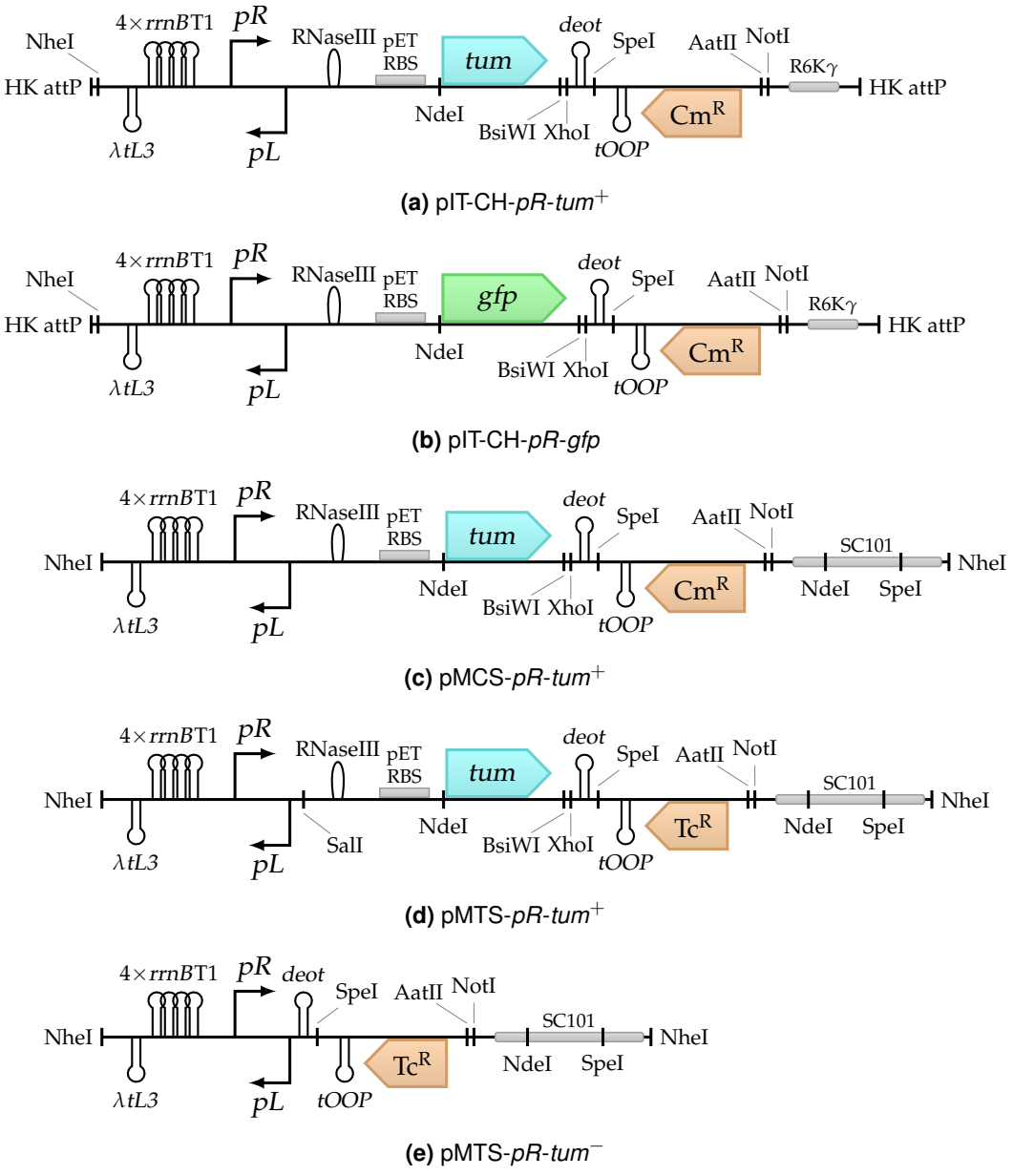
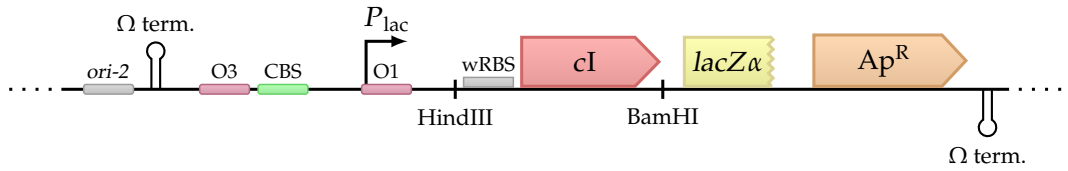
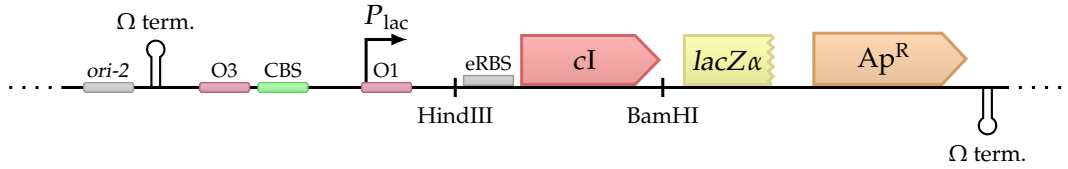


Figure 3.3: Sequence maps of the plasmids used for adding a fluorescent reporter to the MFL. The maps illustrate sequence features (not to scale) for each of the plasmids as indicated, where the CRIM-based plasmids, (a)–(b), are shown linearised at the HK022 attP (HK attP) sites as they would appear when integrated in the host chromosome, and the medium-copy plasmids, (c)–(e), are shown linearised at the NheI restriction enzyme site; low-copy variants of these plasmids with the SC101* origin were also prepared. Features are as described in Figure 3.2 with the additions that *gfp* is the gene for folding reporter green fluorescent protein and $Tc^R$is a gene conferring tetracycline resistance. **(a)** Precursor to **(b)** the integrable *pR-gfp* reporter module; **(c)** precursor to **(d)** the tetracycline resistant version of the MFL plasmid and precursor to **(e)** the *tum*$^-$ control plasmid.



(a) pZC320-WR-*c*I



(b) pZC320-ER-*c*I

Figure 3.4: Sequence maps of the CI expression plasmids. The maps illustrate sequence features (not to scale) around the multiple cloning site of the pZC320 plasmid; a majority of the plasmid is taken up by the genes required for plasmid maintenance at close to chromosomal copy-numbers. Features are as described in Figure 3.2 with the additions that *c*I is the wild-type 186 repressor gene, with either the wild-type RBS (wRBS) or enhanced RBS (eRBS), and Ap^{R}is a gene conferring ampicillin resistance. The operators O1 and O3 are the sites where *lac* repressor binds to repress the P_{lac} promoter; the CAP Binding Site (CBS) enables CAP binding and activation of the P_{lac} promoter.

difficulties faced in this step, including a thorough verification of the identity of each of the final plasmids, are more completely addressed in Section 8.2.9.

Tum$^-$ controls were prepared from these plasmids by digesting with XhoI and SalI, which have compatible ends, and religating in conditions favouring loss of the insert. This results in complete loss of the *tum* gene, RNaseIII site and RBS as shown in Figure 3.3(e).

This final set of four tetracycline-resistant *pR-tum* plasmids, that is, *tum*$^+$ and *tum*$^-$ variants for both medium-copy (SC101) and low-copy (SC101*) versions, were transformed into the dual reporter strain along with the induction plasmid, pZC320-WR-*c*I.

3.1.3 Shifting the range of CI expression levels

To obtain a complete hysteresis loop like that described in Section 2.4.2, and illustrated in Figure 2.10, both ends of the hysteresis curve should be located in monostable regions. When measuring hysteresis of the MFL with the medium-copy *pR-tum* plasmid (pMCS-*pR-tum*$^+$) as a function of CI production rates, Pocock [2007] observed monostability for the lowest levels of induction (i.e., lowest CI production rates), but for the highest levels of induction, the system still appeared to be operating in a bistable region of the hysteresis curve. With the low-copy *pR-tum* module (pMCS*-*pR-tum*$^+$), both ends of the hysteresis curve appeared monostable, but the size of the bistable region was so reduced as to be undetectable within experimental error.

Re-optimising the *pR-tum* module with a production rate between that of the low- and medium-copy variants could have been tried. However, a simpler alternative was to increase the range of CI production rates accessible to the induction system by replacing the

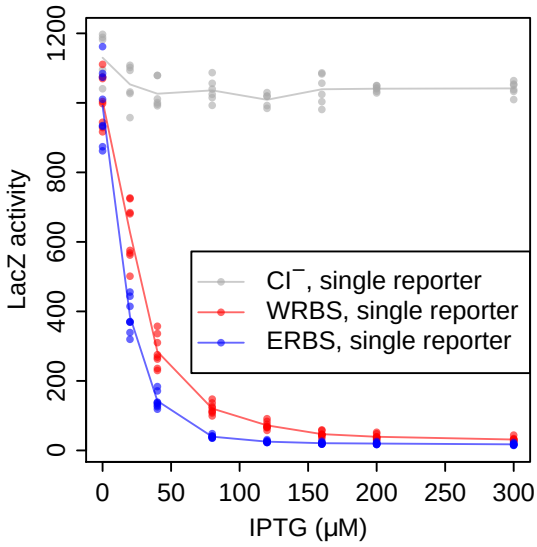


Figure 3.5: Comparing repression of the *pR* promoter by CI expression plasmids that utilise alternative *c*I RBSs. The activity of the *pR* promoter was measured by LacZ assay for the (single reporter) E4300 (pIT-SL-*pR-lacZ*)$_\lambda$ pUHA-1 strain with either the pZC320 (CI$^-$), pZC320-WR-*c*I (WRBS) or pZC320-ER-*c*I (ERBS) plasmids, grown in media with a range of IPTG concentrations to induce expression of the CI repressor protein. For the WRBS and ERBS curves, 8 replicates were assayed at each final IPTG concentration, whilst for the CI$^-$ control, only 6 replicates were assayed.

weaker wild-type RBS (wRBS) for *c*I with the stronger RBS of the pET vectors. A *c*I template with this enhanced RBS (eRBS) was available from the pET3a-*c*I plasmid [Shearwin and Egan, 1996], and this was used to replace the wRBS and *c*I gene between the HindIII and BamHI sites (see Figure 3.4).

The increased rate of CI production due to the eRBS was confirmed by observing the increase in repression of *pR* by LacZ assay in the single reporter strain, E4300 (pIT-SL-*pR-lacZ*)$_\lambda$ pUHA-1 pZC320-ER-*c*I (see Figure 3.5). For the same levels of IPTG induction, the reporter strain with the pZC320-ER-*c*I plasmid produced lower LacZ units than the equivalent strain with the pZC320-WR-*c*I plasmid, indicating increased repression at *pR* and implicating increased expression of CI. A CI$^-$ control strain with an empty pZC320 plasmid instead, indicated the maximum production rate from *pR* without repression. This showed that at minimum induction (0 μM IPTG), the wRBS and eRBS variants can both reach levels of CI production low enough that *pR* production is close to that of the fully unrepressed promoter.

With the addition of these two *c*I RBS variants, a total of eight different assay strains were prepared: four MFL strains each with a respective *tum*$^-$ control strain. These strains will be described using 'WR' and 'ER' to refer to wRBS and eRBS CI induction variants, and 'MC' and 'LC' to refer to low-copy and medium-copy *pR-tum* module variants. Written out in full, the final four MFL strains are:

WR-MC E4300DR pZC320-WR-*c*I pMTS-*pR-tum*$^+$,
WR-LC E4300DR pZC320-WR-*c*I pMTS*-*pR-tum*$^+$ (pMTS*-*pR-tum*$^+$),
ER-MC E4300DR pZC320-ER-*c*I pMTS-*pR-tum*$^+$, and
ER-LC E4300DR pZC320-ER-*c*I pMTS*-*pR-tum*$^+$.

3.2 Assaying hysteresis in the Tum−CI Mixed Feedback Loop

As introduced in Chapter 2, the assay for studying bistability in the MFL strains is based on the measurement of a hysteresis loop as the production rate of CI is varied. This is easily achieved in the experimental system by suitably varying the concentration of the IPTG inducer in the growth medium over time. By first growing in media with concentrations of IPTG that put the MFL strains into one of the monostable states, and then growing in media with a range of IPTG concentrations that cover the bistable region, different stable states will be favoured depending on which monostable state the strains first occupied. Ideally, for low enough concentrations of IPTG, the repressed P_{lac} promoter with a low production rate of CI sets the MFL in the low CI, high Tum monostable state (the 'on' state). Conversely, for high enough concentrations of IPTG, where repression of the P_{lac} promoter is relieved, the consequent high production rate of CI ideally sets the MFL in the high CI, low Tum monostable state (the 'off' state). A crucial requirement of this assay of hysteresis is the ability to 'set the switch' in both on and off states by changing the concentration of IPTG.

Induction of the P_{lac} promoter by IPTG is sigmoidal [Palmer et al., 2009]; depending on the LacI repressor concentration, there will be leaky production from P_{lac} even in the absence of IPTG, but also saturation towards some maximum production rate where almost all LacI is bound (and inactivated) by IPTG. To maximise the chance of starting in a monostable region, the lowest and highest possible production rates of CI should be chosen. The lowest possible production rate is obtained with no IPTG, but some small rate of production from the repressed promoter should be expected. The production rate of CI saturates for increasing IPTG concentrations, which sets the upper limit for CI production, and for very high concentrations of IPTG, increases in inducer concentration produce minimal gains in production rate. Such high concentrations of IPTG are also more likely to adversely affect cell health, so a moderately high choice for the maximum IPTG concentration is preferred. With these considerations in mind, the minimum and maximum induction levels were set at 0 μM and 300 μM IPTG [Pocock, 2007].

A standard LacZ assay [Dodd et al., 2001; Palmer et al., 2009] involves: (1) growing cultures overnight in LB, each innoculated from independent colonies on a streak plate, (2) normalising these cultures to a common OD_{600} (diluting approximately 1/10 in the process), (3) subculturing 1/50 into LB for growth to log phase (OD_{600} 0.65–0.75 [Palmer et al., 2009]), and (4) assaying the log phase cultures for β-gal activity [Dodd et al., 2001]. To tailor this protocol for measuring hysteresis, Pocock [2007] used overnight (O/N) growth for setting the switch by preparing two cultures from each colony, one with 0 μM IPTG (the low O/N) and the other with 300 μM IPTG (the high O/N). After overnight growth, these cultures could each be independently normalised, and then subcultured into LB broth with a range of IPTG concentrations to assay the effect of hysteresis in the bistable region. With the assay as stated, Pocock found that the time taken for the cultures to reach log

phase did not seem to be sufficient for reaching equilibrium conditions at the new IPTG concentrations.

The importance of the time taken to reach equilibrium, the 'equilibration time', has already been highlighted in this thesis in the deterministic modelling of Section 2.4.2. To best judge the boundaries of the bistable region, a longer equilibration time is preferred, though 'perfect' boundaries can never be obtained in finite time. It might at first seem that just leaving the cultures growing longer would solve the problem. However, an important consideration is the growth phase of the culture: when the cell density of the growing culture becomes too high, the growth rate slows down until the cells reach stationary phase and stop dividing. The transition to stationary phase is highly complex, and includes the increased expression of an alternative sigma factor for RNA polymerase (RNAP), which radically alters the pattern of gene expression [Battesti et al., 2011]; such gross cellular changes would obfuscate the results and analysis. In contrast, by including an additional step of dilution before subculturing into LB, the cells start at a lower concentration and hence take longer to reach the same cell densities. Pocock [2007] trialled such an approach, and here an additional 1/10 culture dilution step between the normalisation and subculturing steps is chosen as the standard for obtaining a 'normal' equilibration time (typically around 6 hours). Without this additional 1/10 dilution, the assay will be referred to as having a 'short' equilibration time (typically around 5 hours).

Hence, the primary assay of hysteresis used in this thesis can be summarised:

1. Grow cultures overnight in LB containing either 0 or 300 μM IPTG (low or high O/Ns respectively) to set the switch in one of the monostable regions,
2. Normalise the cultures to a common OD_{600} of 0.15 (diluting approximately 1/10 in the process),
3. Further dilute these normalised cultures 1/10,
4. Subculture 1/50 into intermediate IPTG concentrations in the bistable region and grow to log phase,
5. Measure the production rate from *pR* by LacZ assay or flow cytometry.

In this version of the assay, log phase growth is started from a culture with an overall dilution of approximately 1/5000 relative to the original overnight culture, compared with a dilution of approximately 1/500 for the version with short equilibration time. For technical details of the protocol, see the Materials and Methods (Chapter 8).

The four dual reporter MFL strains and their respective *tum*$^-$ controls were each assayed for hysteresis by LacZ assay using a normal equilibration time, and the resulting curves are shown plotted in Figure 3.6. In all cases, there are points of significant separation between the low O/N and high O/N MFL curves, though the magnitude of this separation varies considerably between each case. This is evidence that all the strains are exhibiting some level of hysteresis (history-dependent behaviour), and hence implies that each strain contains a region of bistability at intermediate IPTG concentrations.

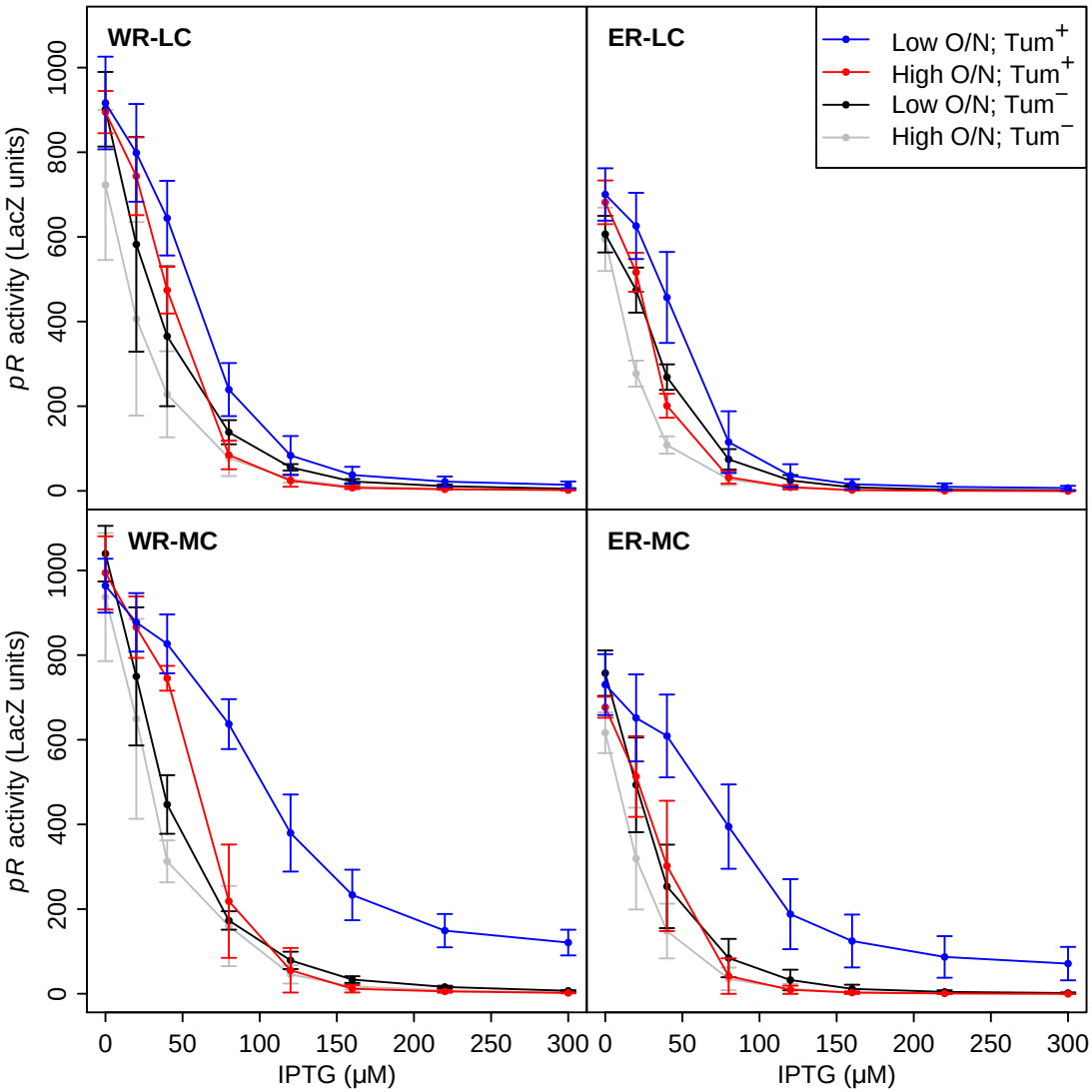


Figure 3.6: The Tum−CI MFL displays hysteresis as the production rate of CI is varied. Hysteresis assays were performed using a normal equilibration time (see text for details), and the activity of the *pR* promoter was measured by LacZ assay. The assay was performed for four different MFL strains as labelled; translation of CI was either from its wild-type RBS (WR) or enhanced RBS (ER), and the *pR-tum* module was either located on a low-copy plasmid (LC) or medium-copy plasmid (MC). Assays of the respective Tum^- control strains are also shown. For each strain, LacZ assays were performed on log phase cultures innoculated either with overnight (O/N) cultures grown in the presence of 0 μM IPTG (low O/N curves), or in the presence of 300 μM IPTG (High O/N curves). Error bars show 95% confidence limits in the mean; $n = 6$ for all data points except for the high overnight Tum^+ curve of the ER-LC strain, which has $n = 4$ for all data points.

Further confirmation of hysteretic behaviour is found in the distinct behaviour of the MFL strains compared with the Tum$^-$ controls, which should be incapable of supporting bistability. However, in some cases, the controls also show significant separation between low and high O/N curves. This could be mistaken for hysteresis, but is more likely an indication that equilibrium is not yet reached (compare with the predicted separation between the control curves from the modelling in Section 2.4.2). This indicates that even with the additional dilution step, there is still insufficient time for CI and β-gal concentrations to reach their steady-state levels. This implies that the MFL curves have not yet reached equilibrium either, and are probably further from equilibrium than the controls, since the positive feedback loop due to Tum was predicted to slow equilibration in the deterministic model. Modifications to the assay protocol to further extend the equilibration time will be considered in Chapter 4.

The key differences between each of the MFL strains involve changes to the production rates from *pR* and P_{lac}. These bring about changes in the shape of the hysteresis curves with the largest changes being observed between strains with low-copy versus medium-copy *pR-tum* plasmids. As anticipated from the modelling, increasing the production rate of Tum, in this case by increasing the copy number of the *pR-tum* plasmid, causes an increase in the size of the bistable region, increasing the point of bifurcation for the low O/N curve in particular (compare with the plot in Figure 2.9(a), Section 2.3.2, of the predicted change in position of bifurcation points for changes to the effective Tum production rate). The result is that the separation between low and high O/N curves is most obvious for the WR-MC and ER-MC strains.

However, it is alarming that for these strains with medium-copy *pR-tum* plasmids, a complete hysteresis loop is not observed, raising the question of how both halves of the hysteresis loop are accessed. The loop is broken by the apparent absence of a single monostable state with high CI and low Tum at the maximum induction level of 300 μM IPTG. This could simply be a result of the curves remaining out of equilibrium at that point, obscuring an underlying monostable regime. Early attempts at further extending the equilibration time for the WR-MC strain, however, still did not see these two curves brought together at the maximum induction level [Pocock, 2007]. So at least in the case of the WR-MC strain, a more likely explanation could be drawn from the differences in circuit parameters between cultures in log-phase (used to assay the switch) and those in stationary phase (used to set the switch). In stationary phase, the slowing of cell division may result in unbalanced levels of CI and Tum that disproportionately favour high levels of CI. Alternatively, since stationary phase appears to result in much noisier gene expression [Guido et al., 2006], this additional noise may provide a pathway for O/N cultures grown at 300 μM IPTG to reach the low Tum state even if the MFL is bistable at that point. This last explanation presumes that the low Tum state is the more stable one at that level of induction, which could be argued on the basis of how close it appears to be to the region

of monostability.

The enhanced RBS for production of CI was designed to increase the overall production rate of CI at all concentrations of IPTG in an attempt to reach the high CI monostable state in log phase cultures. However, even with this increased production rate, the high CI monostable state was still not apparent in the hysteresis assay shown in Figure 3.6. Nonetheless, the separation was reduced by almost a half, demonstrating that the eRBS had an effect. The effect of the eRBS is easier to see at lower concentrations of IPTG. At 0 μM IPTG, the eRBS strains show significantly lower activity compared with the wRBS strains, indicating that leaky transcription from P_{lac} in the absence of induction is also amplified by the stronger RBS, resulting in more CI and hence additional repression of *pR* at this level of induction.

Interestingly, the higher levels of CI production also appear to increase the time taken to reach equilibrium, since the Tum$^-$ control curves show slightly more separation in the eRBS strains compared with the wRBS strains. A potential explanation may be that it takes longer to reach equilibrium when starting from the high levels of CI in the high O/N curves. A deeper understanding of this observation will require better characterisation of the induction module (Sections 3.3 and 3.5.4) and a quantitative comparison with the model (Chapter 4).

In contrast with the hysteresis curves predicted by the modelling, with close to a rectangular shape for the bistable region, the experimental hysteresis curves in Figure 3.6 appear smoothed out or skewed, thereby obscuring the distinct thresholds expected at the points of bifurcation (the IPTG concentrations near which the system transitions between bistability and monostability). This poorly defined shape could indicate a major deficiency in the model, but is more likely a product of inhomogeneity within the cell population. Genetic circuits are inherently noisy [Elowitz et al., 2002] and this stochastic element is likely to impact the results of the MFL assay in a number of ways. One source of inhomogeneity would be variation in the extent of equilibration between cells at the time of assay — the cells in the starting population for the growth to log phase cannot be expected to be synchronised or to each contain equal starting concentrations of CI and Tum. Variations in growth rate between cells during the course of the assay would also decrease equilibration homogeneity. Another source of population inhomogeneity could well be a mixed population of cells in the on state and cells in the off state. In the LacZ assay, this would be measured as an average activity weighted by the number of cells in each population. Distinguishing between these alternative sources of inhomogeneity, or ruling out inhomogeneity as an explanation for the results is beyond the capabilities of the LacZ assay, and will be better addressed by fluorescence-based assays in subsequent chapters.

A number of deficiencies in interpreting these preliminary results within the current framework have been highlighted in this section. Such an outcome is not entirely undesirable, but motivates quantitative characterisation of the constituent circuit components

and a rethinking of core design principles, which thus satisfies a major goal of synthetic biology in better understanding gene network design [Sprinzak and Elowitz, 2005; Yeh and Lim, 2007].

3.3 Characterising the CI induction module

The assay curves presented in the previous section showed the hysteretic response of the Tum−CI MFL genetic circuit to changes in the concentration of IPTG in the growth medium. In contrast, the hysteresis curves predicted by the MFL model derived in Chapter 2 were presented in terms of the production rate of CI. The concentration of IPTG affects the rate of production of CI repressor from the P_{lac} promoter by relieving repression due to LacI repressor. The assumption that this production rate from P_{lac} is proportional to the concentration of IPTG is often good enough for a qualitative analysis, like that of the previous section, but such an assumption masks what is actually a nonlinear response. LacI binds cooperatively to P_{lac} as a tetramer [Oehler et al., 1990], and indeed, the production of downstream genes has been observed to display an ultrasensitive dependence on the concentration of IPTG [Palmer et al., 2009]. This nonlinear dependence also arises as a result of the limited number of LacI repressor molecules per cell, such that the rate of CI production saturates as IPTG reaches concentrations that can inactivate the available pool of LacI. Since variations in the production rate of CI are fundamental to the hysteresis assay, better quantifying the dependence of this production rate on the concentration of IPTG would greatly improve comparability between the results and modelling.

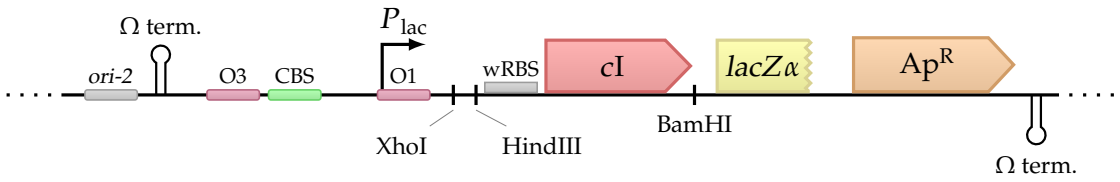
The form of the nonlinearity is likely to be strain- and construct-dependent, since changes to growth rate or copy number could affect both the concentration of LacI repressor and the response at the promoter. This means that the dependence of production rate on IPTG concentration should be determined in strains and assays that closely represent the MFL hysteresis curves. One of the most direct measures, and that employed by Palmer et al. [2009], is to quantitate the steady-state concentration of induced protein (that is, CI in the case of the MFL) in the assay strains for a series of IPTG concentrations using a technique such as Western blotting. The pZC320-WR-*c*I induction curve was previously calibrated for assays of *pR* and *pL* activity by Western blot for induction levels between 10 and 100 μM [Dodd and Egan, 2002]. However, whilst direct, Western blotting tends to afford limited sensitivity and less flexibility; ideally the production rate for each IPTG concentration used in the hysteresis assay would be measured. So here instead, an alternative approach is taken, whereby a *lacZ* reporter gene is used to report on production from P_{lac}. In this approach, production of the β-gal enzyme acts as a proxy for production of CI; this method is less direct, but a LacZ assay affords more sensitivity and enables closer consistency with the hystersis assay.

In order for the production of β-gal to be valuable as a reporter for the production

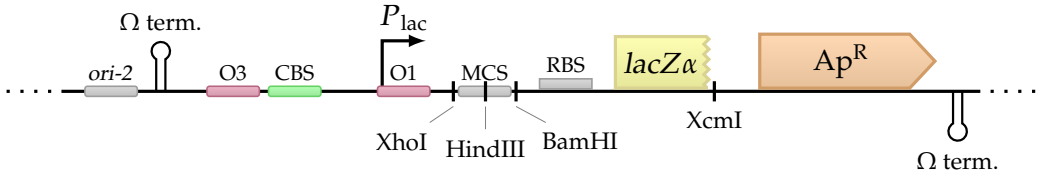
of CI, the steps in its production (and even degradation) must be related in a predictable way (preferably by a proportionality) to that of CI in the MFL. Such differences will be discussed in the course of describing the induction reporter strain design. However, more generally there are still a number of potential drawbacks to this approach that are worth bearing in mind. Firstly, since the *lacZ* gene is already employed in the MFL strains to report on *pR* activity, an alternative strain design must be used. However, if all plasmids and integrands are kept consistent between the strains, excepting removal of the *pR-lacZ* reporter and replacement of the *c*I gene with *lacZ*, then any differences in strain physiology should be sufficiently minimal. A second drawback is that the use of a LacZ assay does not provide an absolute measure of CI production rate — the desired quantity for comparison with the modelling. Nonetheless, the LacZ assay does provide a relative measure of production rate, which can be used to normalise the IPTG concentrations to a quantity that is proportional to the CI production rate. Then by measuring an absolute CI production rate for one of the IPTG concentrations, the relation can be scaled to reproduce the absolute values. Such a reference absolute production rate will be quantified later in this chapter. Finally, by changing the gene downstream of the P_{lac} promoter, changes to the dynamics of target gene translation, and hence its production rate, are unavoidable. Even the strength of the RBS is impacted by changes in the first 35 base pairs of the translated gene [Kudla et al., 2009]. However, if it is assumed that the rate-limiting step in production of the target gene(s) is transcription initiation, as assumed in the modelling in Section 2.2.3, then the rate of transcription initiation becomes the most important factor in determining the response of protein production to changes in repressor occupancy. This assumption presumes that all intermediate steps of protein production that occur after transcription initiation, whether for CI or β-gal, can be considered first order processes, operating at maximal efficiency for every rate of transcription initiation considered.

The P_{lac}-*lacZ* reporter construct was designed to minimise sequence differences with the MFL induction plasmid, pZC320-WR-*c*I, but some modifications were necessary to preserve plasmid integrity and reporter consistency. The pZC320 backbone (see the sequence map in Figure 3.7(b)) contains *lacZα*, a gene for the β-galα fragment of β-gal. When expressed with its complement, the β-galω fragment, the β-galα fragment can pair to produce a functional β-gal enzyme, making it useful as a cloning marker for distinguishing between clones with and without the fragment [Shi and Biek, 1995]. The CI expression plasmid was cloned without replacement of this *lacZα* gene (see the sequence map in Figure 3.7(a)). However, since *lacZα* is identical to the start of the *lacZ* gene, if it was also retained in the induction reporter plasmid, then the reliability of the reporter construct could be compromised by spurious homologous recombination within the plasmid. To avoid such complications, the first cloning step involved removal of the *lacZα* fragment from pZC320 before introducing the *lacZ* gene.

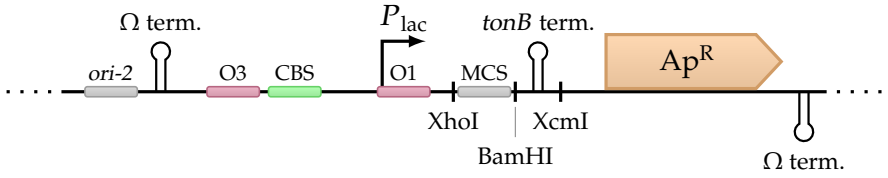
Removal of *lacZα* was achieved by replacing it with the *tonB* terminator between



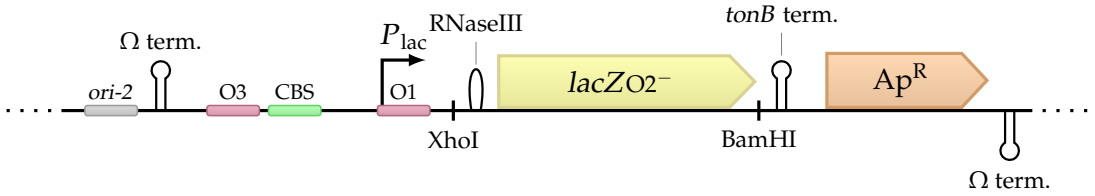
(a) pZC320-WR-*cI*



(b) pZC320



(c) pZC320-*tonBterm*



(d) pZC320t-*lacZ*

Figure 3.7: Sequence maps illustrating plasmid precursors to the IPTG induction reporter plasmid, in which expression of the β-gal enzyme is used as a proxy for expression of CI, so that the ultrasensitivity of the induction system can be quantitated by LacZ assay. The design attempts to replicate **(a)** the CI expression plasmid of the MFL as far as possible. However, to avoid the potential loss of the *lacZ* reporter gene by recombination with the *lacZ*α fragment, this fragment was first replaced in **(b)** the backbone vector with a terminator to create **(c)** the pZC320-*tonBterm* plasmid. **(d)** The *lacZ*O2$^-$ gene, which has a silent mutation to inactivate the internal *lac* repressor O2 binding site, was then cloned into the multiple cloning site (MCS) to create the induction reporter plasmid pZC320t-*lacZ*. The maps illustrate sequence features (not to scale) in and around the multiple cloning site of the pZC320 plasmid; a majority of the plasmid is taken up by the genes required for plasmid maintenance at close to chromosomal copy-numbers. Features are as described in Figure 3.2 with the additions that *cI* is the wild-type 186 repressor gene with its wild-type RBS (wRBS), and Ap^{R}is a gene conferring ampicillin resistance. The operators O1 and O3 are the sites where *lac* repressor binds to repress the P_{lac} promoter; the CAP Binding Site (CBS) enables CAP binding and activation of the P_{lac} promoter.

the BamHI and XcmI restriction sites to create the pZC320-*tonBterm* plasmid (see Figure 3.7(c)). The pZC320 plasmid contains no obvious transcriptional terminators downstream of the *lacZα* promoter, whilst the reporter constructs used elsewhere do typically include a downstream terminator [Simons et al., 1987; Linn and St Pierre, 1990; Dodd and Egan, 2002, and Section 3.1]. Its inclusion should reduce context-dependent effects on mRNA stability resulting from the 3′ end of the transcript [Belasco, 2010], thereby increasing consistency in LacZ units between different reporter constructs. However, it also introduces an additional level of discrepancy with the pZC320-WR-cI plasmid; strong promoters have been known to affect plasmid copy number when incident over the origin [Stueber and Bujard, 1982]. With such a large fraction of the plasmid responsible for copy-number maintenance, and a downstream ampicillin resistance gene in any case, this potentiality was neglected. The transcriptional terminator was introduced in this step to allow for greater freedom in choosing restriction sites to clone *lacZ* upstream.

The wild-type *lacZ* gene contains an internal LacI repressor binding site, O2, which contributes to repressor cooperativity at the P_{lac} promoter [Oehler et al., 1990]. Since such a distal LacI binding site is not present in the CI induction module, it was important to choose a *lacZ* reporter gene in which this site had been silently inactivated. The *lacZ* gene in the pIT3-CL-*lacZ*flip plasmid[2] contains the O2^{-} mutation of Oehler et al. [1990] and was cloned into pZC320-*tonBterm* between the XhoI and BamHI restriction sites to make the induction reporter plasmid, pZC320t-*lacZ* (see Figure 3.7(d)). To further increase consistency of this reporter construct with other measurements of promoter strength, the chosen *lacZ* fragment also contained an RNaseIII site to ensure that upstream sequences would not contribute to differences in stability of the *lacZ* mRNA or translation of the gene [Linn and St Pierre, 1990].

In order to match the MFL strains as closely as possible, the induction reporter strains were kept as similar as possible to the dual reporter MFL strains. So that the only source of β-gal was from the induction plasmid, the *pR-lacZ* reporter of the MFL, (pIT-SL-*pR-lacZ*)$_{\lambda}$, was replaced with (pIT-SL)$_{\lambda}$. pIT-SL was created from placatt1-Δ*lacY-lacZ* (for the sequence map, refer back to Figure 3.2(a)) by digesting with XhoI and SalI, which have compatible ends, and religating under conditions favouring loss of the insert. With this modification of the base strain, two alternative induction reporter strains were cloned:

- E4300 (pIT-SL)$_{\lambda}$ (pIT-CH-*pR-gfp*)$_{\mathrm{HK}}$ pUHA-1 pZC320t-*lacZ* pMTS-*pR-tum*$^{+}$, and
- E4300 (pIT-SL)$_{\lambda}$ (pIT-CH-*pR-gfp*)$_{\mathrm{HK}}$ pUHA-1 pZC320t-*lacZ* pMTS-*pR-tum*$^{-}$.

These induction strains were assayed for β-gal activity using the same hysteresis protocol as for the MFL strains (described in Section 3.2). Briefly, cultures were started from

[2] pIT3-CL-*lacZ*flip [Ian Dodd, unpublished data] is similar to the pIT3-HFCL plasmid described in Cui et al. [2013].

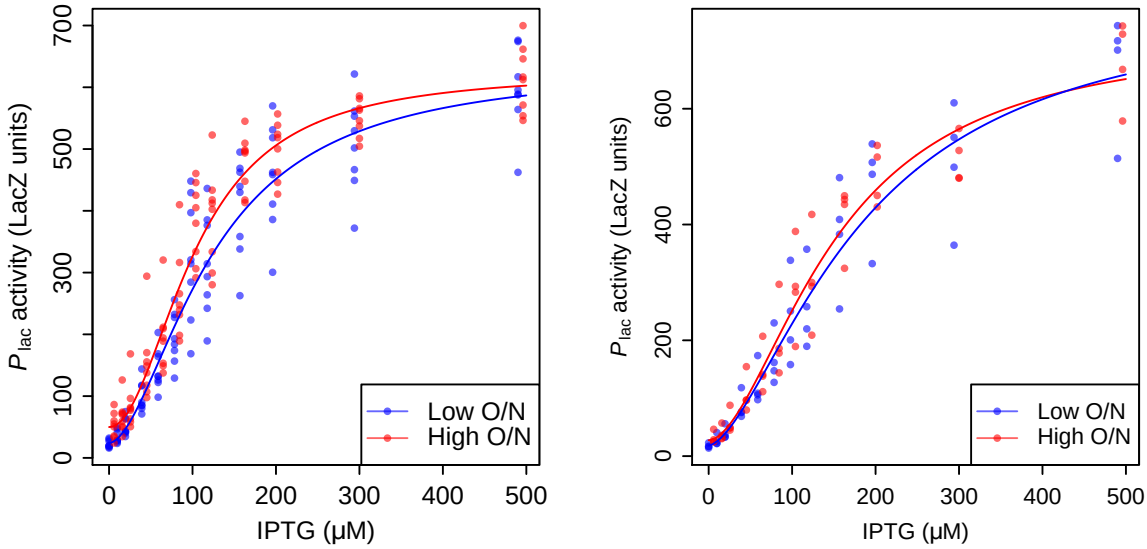


Figure 3.8: Induction of the P_{lac} promoter has an ultrasensitive dependence on the concentration of IPTG inducer in an MFL-like strain. P_{lac} promoter activity was measured by LacZ assay in the induction reporter strains **(a)** E4300 (pIT-SL)$_{\lambda}$ (pIT-CH-*pR-gfp*)$_{HK}$ pUHA-1 pZC320t-*lacZ* pMTS-*pR-tum*$^{+}$, or **(b)** E4300 (pIT-SL)$_{\lambda}$ (pIT-CH-*pR-gfp*)$_{HK}$ pUHA-1 pZC320t-*lacZ* pMTS-*pR-tum*$^{-}$. The assay was based on the hysteresis assay used for the MFL strains, so alternative curves were obtained for low or high O/Ns (as described in Figure 3.6). The IPTG concentration of each data point has been corrected for the different contributions made when subculturing from each respective O/N broth (see text for details). Repeats are plotted as separate data points; $n = 8$ for each condition in (a), whilst $n = 4$ for each condition in (b). The response of each data set was fit using a four-parameter log-logistic function (solid lines); parameters for the fits are given in Table 3.1. The data was fit after Box-Cox transformation with parameter $\lambda = 1/3$.

either low or high IPTG overnights and the additional 1/10 dilution step required for a normal equilibration time was also employed. The resulting two data sets for the *tum*$^{+}$ and *tum*$^{-}$ strains can be seen plotted in Figure 3.8.

An incidental consequence of following the hysteresis protocol was that the final concentration of IPTG in the log phase cultures was notably different depending upon whether they had been innoculated with low or high O/Ns. The discrepancy arises since the overnight cultures are diluted into broth with the same IPTG concentration, so that 2 μL out of the 100 μL of broth for growth to log phase contains IPTG at the concentrations of the overnight cultures. Thus the final IPTG concentration in terms of the intended assay concentration, x, depends upon whether the overnight culture had low (I_{L}) or high (I_{H}) IPTG, and can be calculated as:

$$\begin{aligned} I_{\mathrm{L}}(x) &= \frac{2}{100} \times (0\ \mu\mathrm{M}) + \frac{98}{100} \times (x\ \mu\mathrm{M}) \\ I_{\mathrm{H}}(x) &= \frac{2}{100} \times (300\ \mu\mathrm{M}) + \frac{98}{100} \times (x\ \mu\mathrm{M}) \end{aligned} \tag{3.1}$$

The greatest percentage deviation occurs for the cultures from high IPTG, subcultured into low IPTGs. For example, subculturing into 0 or 10 μM IPTG gives $I_{\mathrm{H}}(0) = 6$ μM and $I_{\mathrm{H}}(10) = 15.8$ μM. In contrast, subculturing from low IPTG overnight cultures has a

maximum deviation at $I_{\mathrm{L}}(500) = 490$ μM, which is a much smaller percentage difference. These functions were used in Figure 3.8 to correct the IPTG concentration for each data point and will also be applied for presentation of all subsequent data derived from the hysteresis assay.

With this correction in place, it can be seen from the moderate overlap of the low O/N and high O/N data in Figure 3.8, that production of β-gal has essentially reached its equilibrium point by the time of the assay. β-gal is expected to follow dilution-limited time evolution, thus representing the longest time it would take to reach equilibrium in rapidly dividing cells. So by inference the concentrations of CI in the hysteresis assays of the *tum*$^-$ MFL strains would be expected to have reached equilibrium at least some time during the course of the assay with normal equilibrium time. Thus the disequilibrium of the MFL *tum*$^-$ curves in Figure 3.6 (Section 3.2) likely reflect the additional time it takes for a second dilution limited process (production of β-gal from *pR*) to reach equilibrium.

Curiously, whilst the data from low and high O/Ns present a similar shape, in Figure 3.8(a) in particular, the spread in data points is markedly different for each of the starting conditions. The low O/N data shows far less variation in LacZ units at 0 μM IPTG compared with the high O/N data, and vice-versa for the high O/N data at 300 μM IPTG. This indicates that the further from its initial equilibrium state that the circuit has to travel, the noisier the resulting output, perhaps suggesting substantial variation in equilibrium times, or growth rates between different repeats.

For easier comparison and extrapolation to intermediate values, the data was fit using four-parameter log-logistic curves, also known as Hill curves, given by

$$f(x) = B + \frac{M - B}{1 + \exp\left\{-H\big(\log(x) - \log(\varepsilon)\big)\right\}} = B + \frac{M - B}{1 + \left(\frac{\varepsilon}{x}\right)^H} \tag{3.2}$$

where B is the basal value that occurs at $x = 0$, M is the saturation maximum, H is the Hill coefficient and ε is the EC_{50}. The `drc` package [Ritz and Streibig, 2005], written for the R statistical computing environment [R Development Core Team, 2012], includes non-linear least squares regression algorithms for fitting such curves, and was used to fit the data.

An assumption of regression by non-linear least squares is that all observations are normally distributed and have equal variance. For the induction data, this assumption does not hold: a simple visual inspection of the points reveals that the variance for each experimental condition generally tends to increase for points with higher mean values. Such heteroscedasticity is not unsurprising in this kind of experiment given that a major source of error is expected to arise from variations in gene expression, errors that are expected to accumulate over time as the cells divide and grow. Multiplicative errors like that will result in proportional changes which are often better compared after a log transformation of the data.

However, instead of using a log transformation to minimise heteroscedasticity, a more general Box-Cox transformation of the response was considered. The Box-Cox transfor-

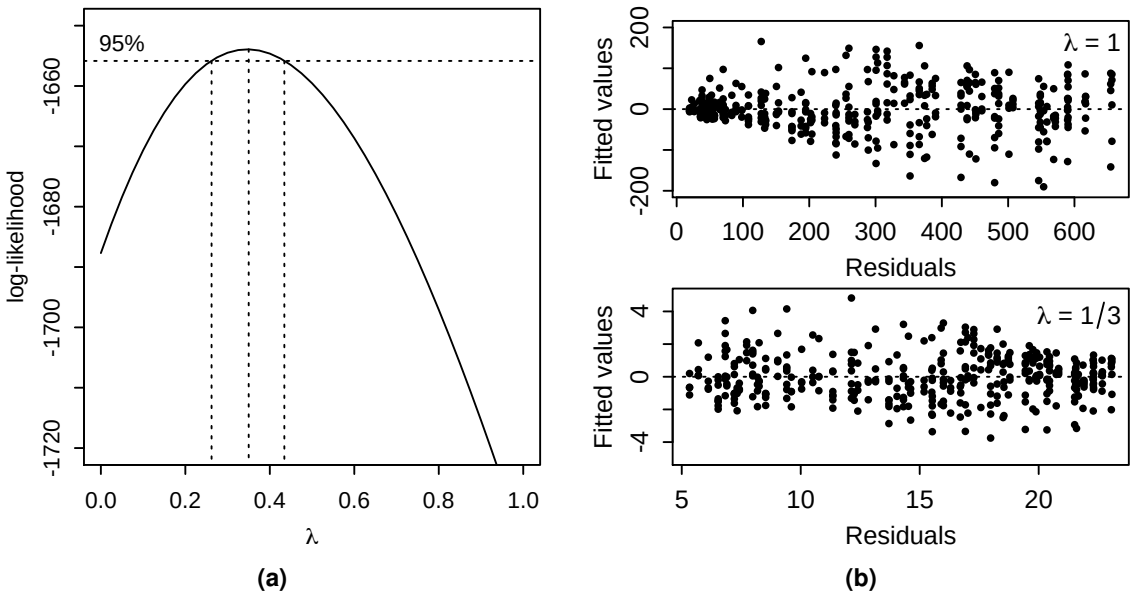


Figure 3.9: Heteroscedasticity is reduced by using a Box-Cox transformation prior to fitting the P_{lac} induction reporter assay data with Hill curves. **(a)** The choice of Box-Cox parameter λ for the transformation was optimised by profiling the log-likelihood for Hill fits of all data sets that P_{lac} activity satisfies a normal error distribution with constant variance over each value of IPTG concentration. The analysis presented here is for the case where the low and high O/N data was treated as an aggregated set in each case. All three relevant data sets can be seen plotted in Figure 3.10. **(b)** The reduction in heteroscedasticity can be seen by comparing plots of residuals versus fitted values taken from each of the fits for different choices of λ. The upper plot with $\lambda = 1$ shows the case where the data is essentially untransformed, and the lower plot with $\lambda = 1/3$ shows the case the data is essentially transformed by a cube root.

mation is:

$$y^{(\lambda)} = \begin{cases} \dfrac{y^{\lambda} - 1}{\lambda}, & \text{if } \lambda \neq 0, \\ \log(y), & \text{if } \lambda = 0. \end{cases} \tag{3.3}$$

and the choice of the parameter λ can be optimised for least squares regression by profiling the log-likelihood that the dependent variable (P_{lac} activity) satisfies a normal error distribution with constant variance for each value of the independent variable (IPTG concentration) [Box and Cox, 1964]. Such a transformation can normalise the variance in a response-dependent manner, and is only useful if there is an underlying trend in the variance with the magnitude of the measurement. Thus a single value for λ should be chosen to transform all data that can be expected to have the same sources of measurement error. In this case, it is assumed that the measurement error arising from LacZ assays of either of the induction strains will exhibit a similar dependence on measurement magnitude.

The log-likelihood of a normal error distribution with constant variance for Box-Cox transformed Hill fits of all the induction data, including that shown in Figure 3.8 and also that shown later in Figures 3.10(a) and 3.10(b), was determined for a range of choices of λ. This was done alternately for the case where the data from high and low O/Ns were fit independently, or where data from high and low O/Ns was aggregated for each fit.

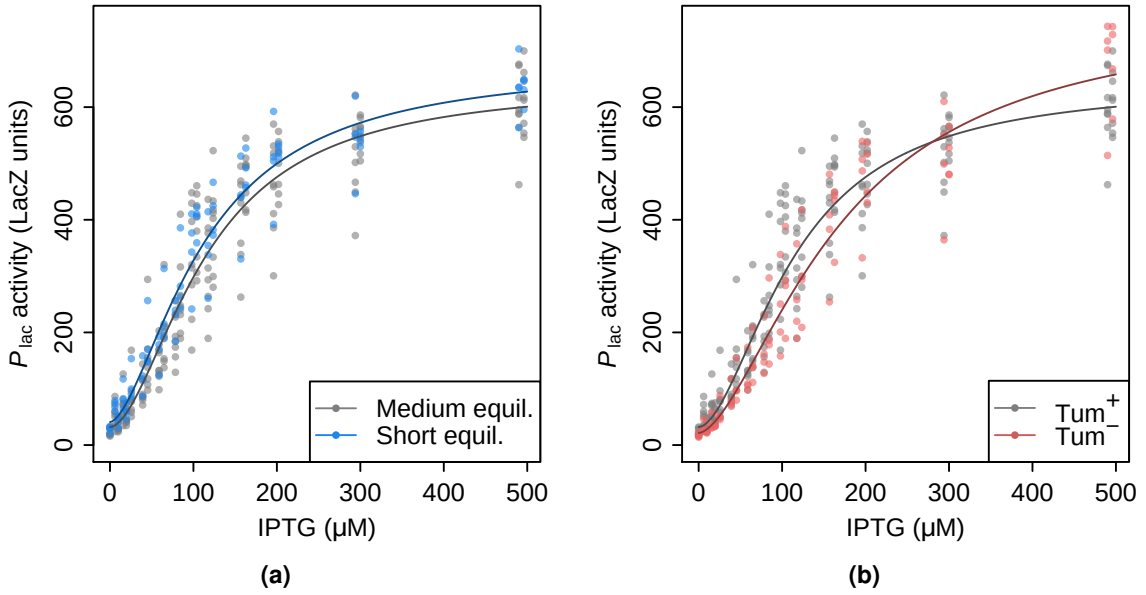


Figure 3.10: Comparing P_{lac} promoter induction under different assay conditions in MFL-like strains. P_{lac} promoter activity was measured as described in Figure 3.8. The Tum$^+$ data from that figure is replicated here in grey for reference, but the high and low O/N data sets have been aggregated. This reference assay, with normal equilibration time (equil.) and the (*pR-tum*$^+$) induction reporter strain, is plotted with **(a)** the *pR-tum*$^+$ induction reporter strain with short equil., or **(b)** the equivalent *pR-tum*$^-$ induction reporter strain with normal equil. Repeats are plotted as separate data points; $n = 4$ for each condition in the short equil. and Tum$^-$ data sets. The data was fit after Box-Cox transformation with parameter $\lambda = 1/3$.

A log-likelihood profile for the latter case (fitting on the aggregated O/N data) is shown in Figure 3.9(a). The maximum occurs at around $\lambda = 0.35$, and from this maximum, the likelihood-ratio test can be used to estimate the 95% confidence limits as 0.26 and 0.43. When the high and low O/N data are fit separately, the estimated maximum becomes instead $\lambda = 0.25$ with 95% confidence limits of 0.15 and 0.33. Taking both λ estimates into consideration, a common optimum λ was chosen to be $1/3$; with $\lambda = 1/3$, the Box-Cox transformation is essentially a cube root transformation.

The effect of this transformation on all the data sets can be gauged from the residuals of the fits which can be plotted against the fitted values to visualise any trends in variance about the fit curves. This is shown in Figure 3.9(b) for the case where the O/N data was aggregated for each fit. Where $\lambda = 1$, essentially the case where there is no transformation, then the data clearly shows a clear trend of larger spread in the residuals for larger fitted values. Where $\lambda = 1/3$, this trend is reduced and the spread in the residuals is more consistent.

The Box-Cox transformation with $\lambda = 1/3$ was applied for producing the fitted curves shown in Figures 3.8 and 3.10(a) and also for determining all of the fitted Hill curve parameters listed in Table 3.1. Application of the Box-Cox transformation makes little difference to these fitted parameter values; the differences are primarily manifest in the error estimates on those parameters, with the largest changes occuring for the basal parameters.

Table 3.1: Parameters determined from Hill fits of the P_{lac}-*lacZ* induction data in MFL-like strains. All data was fit after Box-Cox transformation with $\lambda = 1/3$. The parameter estimates are indicated with 95% confidence limits from the fit.

Strain[a]	Equilibration time[b]	History[c]	Hill coefficient	EC_{50}[d]	Basal[e]	Maximum[e]
tum$^+$	Normal	Low O/N	1.8 ± 0.3	124 ± 24	24 ± 6	635 ± 82
tum$^+$	Normal	High O/N	2.1 ± 0.4	104 ± 15	50 ± 12	625 ± 65
tum$^+$	Short	Low O/N	1.7 ± 0.4	120 ± 34	30 ± 11	679 ± 123
tum$^+$	Short	High O/N	2.0 ± 0.6	99 ± 23	69 ± 21	634 ± 96
tum$^-$	Normal	Low O/N	1.6 ± 0.4	184 ± 67	18 ± 8	787 ± 199
tum$^-$	Normal	High O/N	1.8 ± 0.4	153 ± 46	27 ± 12	728 ± 154
tum$^+$	Normal	Mixed	1.8 ± 0.3	116 ± 17	32 ± 7	642 ± 62
tum$^+$	Short	Mixed	1.7 ± 0.4	113 ± 26	41 ± 12	678 ± 98
tum$^-$	Normal	Mixed	1.6 ± 0.3	170 ± 46	21 ± 7	766 ± 141

[a]Induction reporter strains were either E4300 (pIT-SL)$_\lambda$ (pIT-CH-*pR-gfp*)$_{HK}$ pUHA-1 pZC320t-*lacZ* pMTS-*pR-tum*$^+$, or the *tum*$^-$ equivalent with the pMTS-*pR-tum*$^-$ plasmid instead.

[b]Relative to the short equilibration time assay, the normal equilibration time assay involved an additional 1/10 dilution of the overnight cultures so that the time taken to reach log phase would be longer.

[c]Cultures were initially grown overnight (O/N) in broth with either 0 μM (Low) or 300 μM (High) IPTG; these curves were either fit separately, as indicated, or the fit was for the aggregated dataset (Mixed).

[d]Half-maximal effective concentration; listed in μM.

[e]Listed in LacZ units.

Without the Box-Cox transformation, the 95% confidence estimates on the basal parameters were typically on the order of or, in a number of cases, larger than the values of the parameters themselves.

The Hill fits help to make comparisons between the data sets more objective. The high and low O/N curves shown in Figure 3.8 confirm the similar shapes of the two data sets. Furthermore, from the parameter estimates in Table 3.1 it can be seen that, within 95% confidence limits, all parameters but the basal level of production overlap when comparing high and low O/N curves. This applied even when the equilibration time of the assay was short (that is, omitting the extra 1/10 dilution step of the overnight cultures). The significant difference in the basal levels of production between high and low O/Ns for both short and normal equilibration times presumably arises as a result of the effect mentioned earlier that the further from equilibrium that the system has to travel, the noisier the assay seems to become. In particular, the high O/N data shows much more variation at its lowest level of IPTG than the low O/N data does. This also raises the question of why such a discrepancy between high and low O/N curves does not arise as prominently for the value of the maximum parameter, which should apply for the opposing case where higher variance might be expected in the low O/N data at the highest concentrations of IPTG. However, this may merely be a result of such divergent variances being lost amidst increased levels of assay error associated with higher assay units. Another explanation may also be that a small subset of cells from the high O/N divide much more slowly than the others, thus biasing the results towards higher units overall for that curve.

A comparison between the induction assay with normal equilibration time (here averaging at 6.9 hours) and an induction assay with short equilibration time (5.4 hours) can be found in Figure 3.10(a). High and low O/N data has been aggregated in this case, but clearly the two data sets overlap well. Perhaps surprisingly, even the magnitudes of variance at the lowest concentrations of IPTG appear to be similar, suggesting that the additional variance observed for the high O/N is not just an artefact of limited equilibration. The time difference between the two assays is at least twice as long as the typical doubling time for *E. coli* (30 minutes), so should have been sufficient additional time to at least reduce such discrepancies. So since the differences in variances between low and high O/Ns appeared to have low association with equilibration, it was assumed that such asymmetries in assay error could potentially lead to overestimates of the 'true' mean for a hypothetical ideal assay. For this reason, the aggregated data set including data from both high and low O/Ns was preferred for application in the modelling and in normalising the hysteresis curves.

The induction reporter strain with the pMTS-*pR-tum*$^-$ plasmid was also assayed using the hysteresis assay with normal equilibration time, and that data is shown plotted in Figure 3.10(b). On first glance, the Hill fit of that data appears to have quite a different shape compared with that for the pMTS-*pR-tum*$^+$ plasmid. However, the parameters determined for each fit, as listed in Table 3.1, reveal that, within the 95% confidence limits, there is no significant difference between the two curves. Nonetheless, the range of P_{lac} activities covered by the Tum$^-$ curve is larger, with a noticeably lower basal parameter and higher maximum parameter than the other fits. Furthermore, its EC_{50} is also noticeably larger than those for the other fits. By visual inspection of the data, it could be argued that this may result from the Tum$^-$ data showing a somewhat reduced level of the asymmetric variances observed to result from the history of each data set in the Tum$^+$ strains. An explanation for this is unclear at this stage, and more repeats would be required to show that it is a real difference in any case. However, if it were the case, the Tum$^-$ data would likely serve as the best representative for the steady-state concentrations of CI in the MFL strains.

The Hill curve fits indicate that the nominal choice of 300 μM IPTG as the maximum induction level in the hysteresis assay is a reasonable choice. The fits most closely representing the conditions in the MFL strains, that is, those shown in Figure 3.10(a), reveal that little additional P_{lac} promoter activity is to be gained by almost doubling the inducer concentration to 500 μM. By 300 μM, the curves have well started approaching saturation. The MFL hysteresis curves in Figure 3.6 of the previous section, appeared to indicate that the maximum level of induction was not high enough to reach the high CI monostable state, even with an enhanced RBS for CI production. In this latter case, there is a chance that the slight additional activity afforded by 500 μM could help to tip this end of the MFL into monostability. However, the apparent bistability at that maximum induction level will be

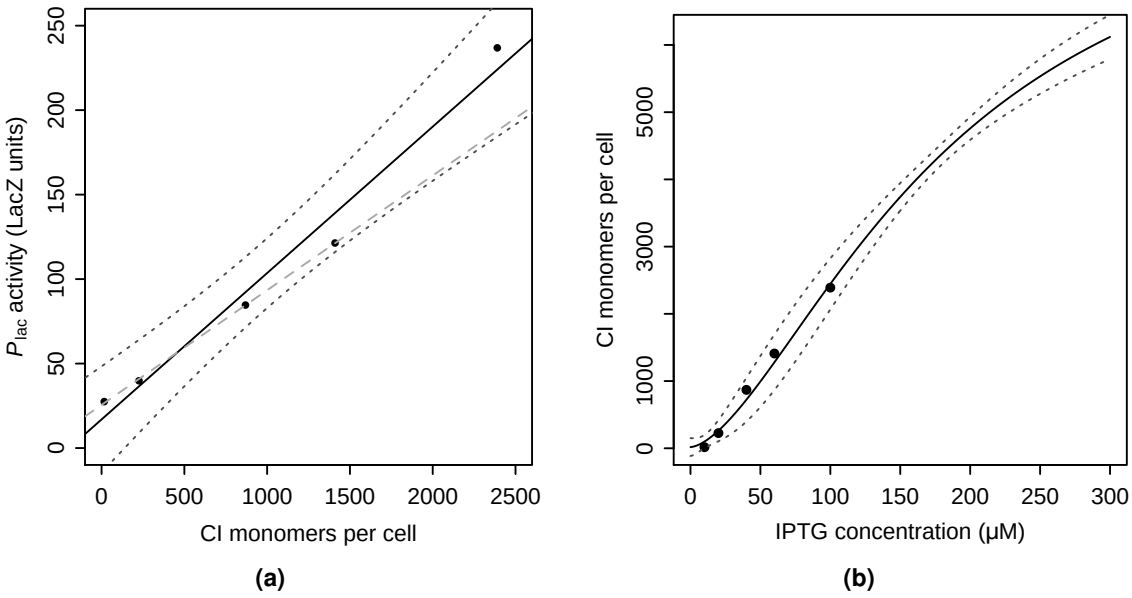


Figure 3.11: Scaling the P_{lac} induction curves measured by LacZ assay to the equivalent steady-state CI concentrations. **(a)** The mean activity of the P_{lac} promoter from the low O/N assay of the Tum$^-$ strain (Figure 3.8(b)) is plot versus measurements of CI concentration per cell [Dodd and Egan, 2002] for induction of P_{lac} from pZC320-WR-*c*I using 10, 20, 40, 60 and 100 μM IPTG. The data was fit by linear regression either including (black solid line) or excluding (grey dashed line) the maximum point. Dotted lines indicate 95% confidence limits on the regression line including all points. **(b)** The linear fit can be used to scale the Hill curve fits of the P_{lac}-*lacZ* induction data to the number of CI monomers per cell. Shown is the scaled low O/N Tum$^-$ Hill curve from Figure 3.8(b) plotted with the measurements of CI concentration per cell from Dodd and Egan [2002]. Dotted lines indicate 95% confidence limits for the fitted Hill curve.

reconciled in other ways later.

The P_{lac}-*lacZ* induction curves have been measured in LacZ units, but for comparison with the modelling, relating these curves to a measure of CI concentration would be more useful. Steady-state CI concentrations had been measured by Western blotting for this induction system previously [Dodd and Egan, 2002], though for a more limited selection of IPTG concentrations. Furthermore, those results were also obtained for a strain (E4300 (λRS45Δ*YA*-pMRR9-HS-$F_L{}^+pL^+pR^+F_R{}^+$) pUHA-1 pZC320-WR-*c*I) whose accompanying reporters and plasmids were somewhat different to those of the MFL. Nonetheless, a comparison of this data with the LacZ assay data would provide a good estimate for the scaling necessary to convert between the two measurements.

Out of the induction reporter strains presented here, the one most closely matching that from Dodd and Egan [2002] would be the *tum*$^-$ variant, since there is no *tum* gene present in that strain. Further, the assay protocol in that paper more closely matches that done for the low O/N. A plot of the low O/N induction Tum$^-$ induction data versus the number of CI monomers per cell as reported in Dodd and Egan [2002], matched up according to IPTG concentration, is shown in Figure 3.11(a). The units of CI concentration are kept as monomers per cell since this avoids the introduction of cell volume estimates, which will be dicussed in Section 3.4.4. The two alternative measurements of induction

from the P_{lac} promoter are well matched by a linear model with $R^2 = 0.98$. The fit has a nonzero y-intercept indicating that there is likely to be some background level of β-gal production that is present irrespective of the level of repression of the P_{lac} promoter. That may indicate that there is an unidentified but weak promoter somewhere downstream of P_{lac} that was introduced along with the *lacZ* gene.

As a brief aside to be returned to later, an alternative fit of the two different sets of induction measurements excluding the maximum data point (for induction at 100 μM IPTG) produces an $R^2 > 0.99$. Blindly excluding such a point without further experimental validation would be highly questionable, but data obtained later in Section 3.5.4 makes use of the extra flexibility such an exclusion affords. The alternative fit lies within the 95% confidence limits of the original in any case. If it were an outlier, the deviation could possibly be explained by a saturation of sensitivity in the Western blotting measurements for its comparatively high level of induction.

The linear model shown in Figure 3.11(a) can be used to scale any of the induction curves measured by LacZ assay to the number of CI monomers per cell. As an example and basic test of this conversion, the model has been used to scale the low O/N Tum$^-$ Hill curve fit from the P_{lac}-*lacZ* data of Figure 3.8(b) to CI monomers per cell. This is displayed along with the measurements of CI from Dodd and Egan [2002] in Figure 3.11(b). The Western blotting measurements of CI concentration all fall within the 95% confidence limits of the Hill curve derived from LacZ assay data. This correspondence of the LacZ assay and Western blotting data will be returned to in Section 3.5.4, where direct measurements in the MFL strains of steady-state CI concentrations are presented.

3.4 Host strain characteristics

The host organism for all the gene networks in this thesis is the well-studied bacterium *E. coli*, and much of the modelling and its predictions relies on the reproducibility and consistency that typically characterise growing cultures of *E. coli*. However, factors such as strain genotype, growth medium and growth temperature can all affect the 'standard' behaviour of *E. coli*. The growth rate is an important indicator for such broad scale changes; changes to the growth rate implicate wide-ranging effects including changes to the rates of loss of dilution limited proteins and even promoter strengths. This section is primarily concerned with the measurement of growth rates for the strains used in this thesis and the consequences that these growth rates have on the modelling, in particular, in setting dilution limited rates of loss and also in determining cell volumes.

3.4.1 Optical density measurements

For measurements of growth rate and also derivations of per cell parameters from bulk culture measurements, a measure of cell concentration is required. The turbidity of a cul-

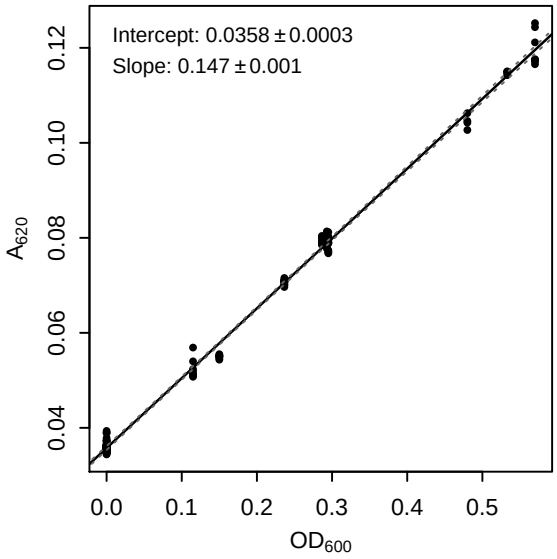


Figure 3.12: Calibrating absorbance measurements at 620 nm, for cultures grown in M9 minimal media in 96-well plates, to standard optical densities at 600 nm (OD_{600}). The OD_{600} was measured for cultures of various densities using an Ultrospec 10 Cell Density Meter, and a concurrent measurement made in 96-well plates using a Multiskan Ascent plate reader. See text for further experimental details. Grey dotted lines mark 99% confidence intervals on the fit. Also indicated are the slope and intercept of the fit with 95% confidence limits.

ture, as measured by the absorbance, varies with the cell mass of the culture, and whilst the relationship is non-linear [Bipatnath et al., 1998], for sufficiently low absorbances a linear approximation is valid. This measure of cell mass, known as the OD_{600}, is typically reported as absorbance at 600 nm using a path length of 1 cm. The cell mass is in turn proportional to the concentration of cells in the culture, but it is important to bear in mind that the constant of proportionality depends on the average mass of each cell in the culture, which may vary between strains and even has a power law dependence on growth rate [Donachie and Robinson, 1987]. Thus where absolute (as opposed to relative) concentrations of cells are required, these need to be calibrated to the OD_{600} by cell counts for each particular assay.

Most of the assays in this thesis are performed with cultures grown in 96-well plates, and measurements of their OD_{600} made using Multiskan Ascent plate readers produced by the Thermo electron corporation[3]. The readers do not have a filter for measuring absorbance at 600 nm, the closest being a filter for 620 nm. The different wavelength affects the effective extinction coefficient, but not the proportionality to cell mass. So with this difference in mind and given that the path length of the measurement is dependent on culture volume in 96-well plates, an empircally-determined linear correction for relating absorbance measurements (absorbance at 620 nm (A_{620})) to OD_{600}s was opted for.

For cells grown in LB this linear correction has already been characterised for 100 µL of culture per well as $OD_{600} = (A_{620} - 0.0394875)\,/0.155$ [Ian Dodd, unpublished data]. For cells grown in M9 minimal media the linear correction was characterised using cultures of E4643 (pITM-*pR-gfp*)$^{\mathrm{Flp}}_{186}$ (pIT3-SH-*lacI*)$_{\mathrm{HK}}$ (pITM-CT-P_{LL5}-*lacZ*)$_{\varphi 21}$ with either pMTS-*pR-tum*$^{+}$ or pMTS-*pR-tum*$^{-}$ plasmids (for the details of these strains, see Chapter 7). Three independent colonies of each strain were grown to stationary-phase in M9 minimal

[3]One of the readers is in fact an older version produced by Labsystems.

media with 20 mM glucose and 4 μg/mL tetracycline, subcultured into the same media and then grown to log-phase. OD_{600}s of the cultures were measured in triplicate using the Ultrospec 10 Cell Density Meter produced by Amersham Biosciences as a standard reference. In particular, measurements were made for the log-phase cultures at an OD_{600} around 0.6, but also for one in five dilutions of the stationary- and log-phase cultures. All these samples were synchronously measured in 96-well plates by pipetting 4×100 μL aliquots of each culture into the wells of the plates.

The resulting data was fit using linear regression (see Figure 3.12) to give $OD_{600} = (A_{620} - 0.0358)/0.147$ for 100 μL aliquots of cultures grown in M9 minimal media. It was also noted that any systematic error observed between the two readers was insignificant in comparison with the random errors introduced elsewhere (like those introduced by pipetting, which affects the path length). Likewise, systematic errors arising between 96-well plates were found to be insignificant.

3.4.2 Growth rate

The growth rate of the cells in a culture has a broad impact on cell morphology and phenotype, affecting properties such as the average cell volume [Donachie and Robinson, 1987], promoter strengths [Liang et al., 1999] and gene copy numbers [Bipatnath et al., 1998]. The effects of growth rate on gene expression in the context of synthetic networks, in particular bistable networks, is considered by Klumpp et al. [2009]. Growth rate modulation by circuit components themselves can even be the cause of bistability in networks with non-cooperative feedback [Tan et al., 2009]. Nonetheless, the impact of growth rate on gene networks and the factors impacting growth rate are still not well understood. Hence, the primary concern here is to control for changes to the growth rate. This can be challenging since the growth rate itself is modulated by cell stresses like the maintenance of high copy or strong promoters (of particular relevance here is the impact of the plasmid-bound *pR* promoter), or additional stresses imposed by antibiotics used for plasmid maintenance. As a result, for the parameter measurements in this chapter, great care is taken to replicate the genotypes and experimental conditions of the assays as closely as possible.

Basic growth rate measurements can be used to track consistency between assays, but an accurate measure of the growth rate also helps to define certain parameters of interest in the modelling. For example, the rate of loss of stable proteins is 'dilution limited', that is, loss occurs primarily as a function of dilution arising from the increase in cell mass/volume of growing cultures. Growth of *E. coli* in LB proceeds exponentially until an OD_{600} of around 0.3 [Sezonov et al., 2007], after which, the rate of growth gradually asymptotes towards a maximum OD_{600} that is characteristic for the given strain and media. However, providing assays are restricted to log-phase (i.e., exponential) growth with growth rate γ, the rate of loss of a dilution limted protein A can be modelled simply as $\frac{\mathrm{d}A}{\mathrm{d}t} = -\gamma A$. Since growth of a culture represents the process of many dividing cells, this parameter is

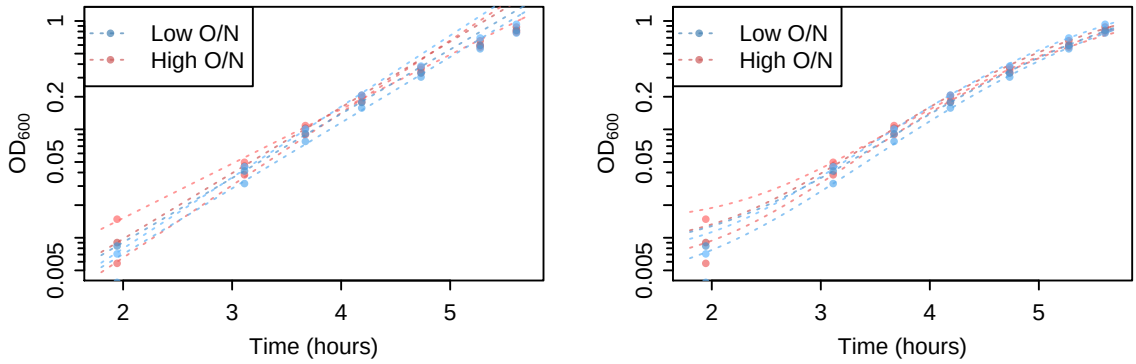
reported as the time it takes for the cell mass to double in size — the doubling time. This is inversely related to the growth rate as determined for exponential growth with base 2 (i.e., doubling time $= \log(2)/\gamma$).

Measurements of growth rate can be made by following the increase in OD_{600} over time of growing cultures using the instruments outlined in Section 3.4.1 and then the growth rate extracted by fitting the resulting curves. Exponential growth is most easily characterised as the rate of *fold* change of the OD_{600}, so that the slope of the log-transformed OD_{600} versus time gives the growth rate at any instant. Where this rate stays constant (i.e., during purely exponential growth), linear regression of the log-transformed data can provide a complete description of the growth rate. However, as mentioned above for cultures growing in LB medium, the rate of growth starts to asymptote above OD_{600} 0.3 and is also normally preceded by a lag time, during which the sensitivity of the instrument is too low to detect the growing cells. Accurately accounting for such effects will require a more sophisticated model of growth.

Zwietering et al. [1990] found that one of the most reliable models for bacterial growth is the Gompertz model and re-formulated this model in terms of parameters with a biological meaning. A detailed description of this model and how it is applied to fit the growth curves in this thesis can be found in Appendix A, but for the purposes here, it is enough to know that the Gompertz model contains a term to account for saturation and another to account for some lag time before growth is observed. The growth rate in the Gompertz model represents the *maximum* rate of growth, which can be determined as the largest slope occurring within the time course of log-transformed OD_{600}s. Three- and four-parameter variants of the Gompertz model are available. With fewer parameters, the three-parameter version is typically preferred, but depending on the number of data points measured inside the lag period, the data may be more accurately modelled by the four-parameter description, which can additionally account for errors in setting background levels of optical density (see Appendix A for a more detailed description).

An example set of time course measurements, in which the optical densities of growing cultures of E4300DR pZC320-ER-*c*I pMTS-*pR-tum*$^+$ were followed, can be found in Figure 3.13. The cultures were grown in 96-well plates, so as mentioned previously in Section 3.4.1, they were first measured as A_{620}s and have been converted to OD_{600}s for the figure. However, since the well-to-well variation measured at the zero time point[4] typically persists over time (see Appendix A), the A_{620} measurement taken at the zero time point has been used as a per-well background reading for the duration of the experiment. This is used instead of the generic background A_{620} reading listed in Section 3.4.1 since it produces a more reliable measure of the relative increases in cell density at low OD_{600}s, but the slope from the empirically-derived correlation between A_{620} and OD_{600} is still used to scale the offset data into an OD_{600}. Such per-well systematic errors are presumably pri-

[4]The zero time point measurement is that taken directly after subculturing for log-phase growth.



(a) Log-linear **(b)** Gompertz

Figure 3.13: Growth curves can be fit **(a)** with a log-linear model by truncating the data using an upper threshold, or **(b)** with a 3 parameter Gompertz model that also accounts for saturation of exponential growth. The A_{620} was followed over time for cultures of E4300DR pZC320-ER-*c*I pMTS-*pR-tum*$^+$ grown in LB with 0 µM IPTG in 96-well plates, but has been converted to an OD_{600} using the empirically derived scaling factor specified in Section 3.4.1. These cultures were started from either low overnight (Low O/N) or high overnight (High O/N) cultures as indicated.

marily a consequence of path length variations arising from pipetting errors.

Another consequence of pipetting errors would be small differences in the absolute number of cells added to each well of the plate at the zero time point. Whilst such differences are amplified exponentially over the course of the assay, they will not affect growth rate determinations since it is the rate of relative change that is important and such differences are expected to be correlated over time in each well. However, this makes it necessary to fit growth curves for each well separately; statistical aggregation can then be performed on these derived growth rates.

To compare the different models of growth, the same example set of growth curves has been fit alternatively using log-linear (Figure 3.13(a)) or Gompertz (Figure 3.13(b)) models. Each of the fits have been transformed back to the original scale and are shown overlaid on the untransformed data. Since the rate of growth slows from purely exponential growth to some saturation point and since the sensitivity of the instrument sets a lower bound on detecting growth, only those data points with OD_{600}s between 0.005 and 0.3 were used in the log-linear fit (the use of these particular thresholds is further discussed and validated in Appendix A). With these bounds in place, it can be seen in Figure 3.13(a) that growth of the cultures deviates quite obviously from the model of pure exponential growth for points with OD_{600}s above 0.4. In contrast, the Gompertz model can well match the slowing rate of growth as seen in Figure 3.13(b). In spite of the significant differences between the two models, the two methods result in very similar doubling time estimates for the curves shown of 30 ± 3 min for the log-linear model and 29 ± 2 min for the Gompertz model. However, growth rates determined using the Gompertz model tend to be more consistent, with reduced variance (see Appendix A). On the whole, log-linear models tend to produce overestimates of the doubling time, since any slowing of growth for OD_{600}s below 0.3 will

Table 3.2: Doubling times measured for notable strains within the present thesis. Growth was followed by monitoring the A_{620} and the resulting curves were fit mainly with 3 parameter Gompertz models as described in the text.

Strain	**Antibiotics**[a]	**Doubling time**[b]	n^c
Basic strains[d]			
E4300 pUHA-1	Kn^{50}	28 ± 2	4
E4300DR[e]	Kn^{50}	26 ± 1	8
E4300 (pIT-SL-*pR-lacZ*)$_\lambda$ pUHA-1 pZC320-ER-*c*I	$Kn^{50}Ap^{30}$	28 ± 4	4
E4300DR pZC320-ER-*c*I	$Kn^{50}Ap^{30}$	27 ± 4	4
Induction reporter strains[f]			
E4300 (pIT-SL)$_\lambda$ (pIT-CH-*pR-gfp*)$_{HK}$ pUHA-1 pZC320t-*lacZ* pMTS-*pR-tum*$^+$	$Kn^{50}Ap^{30}Tc^4$	28.2 ± 0.3	95
E4300 (pIT-SL)$_\lambda$ (pIT-CH-*pR-gfp*)$_{HK}$ pUHA-1 pZC320t-*lacZ* pMTS-*pR-tum*$^-$	$Kn^{50}Ap^{30}Tc^4$	28.9 ± 0.5	96
MFL strains[g]			
E4300DR pZC320-ER-*c*I pMTS-*pR-tum*$^+$	$Kn^{50}Ap^{30}Tc^4$	28.1 ± 0.6	96
E4300DR pZC320-ER-*c*I pMTS-*pR-tum*$^-$	$Kn^{50}Ap^{30}Tc^4$	28.2 ± 0.5	96
E4300DR pZC320-ER-*c*I pMTS*-*pR-tum*$^+$	$Kn^{50}Ap^{30}Tc^4$	29.5 ± 0.5	96
E4300DR pZC320-ER-*c*I pMTS*-*pR-tum*$^-$	$Kn^{50}Ap^{30}Tc^4$	27.9 ± 0.3	96
E4300DR pZC320-WR-*c*I pMTS-*pR-tum*$^+$	$Kn^{50}Ap^{30}Tc^4$	28.3 ± 0.2	96
E4300DR pZC320-WR-*c*I pMTS-*pR-tum*$^-$	$Kn^{50}Ap^{30}Tc^4$	28.2 ± 0.3	95
E4300DR pZC320-WR-*c*I pMTS*-*pR-tum*$^+$	$Kn^{50}Ap^{30}Tc^4$	29.0 ± 0.4	96
E4300DR pZC320-WR-*c*I pMTS*-*pR-tum*$^-$	$Kn^{50}Ap^{30}Tc^4$	29.6 ± 0.4	96

[a] Kn^{50}: 50 μg/mL kanamycin; Ap^{30}: 30 μg/mL ampicillin; Tc^4: 4 μg/mL tetracycline.
[b] Mean doubling time listed in minutes with 95% confidence limits estimated from the set of repeats.
[c] The number of growth curves that were independently measured and fit.
[d] In this group, each growth curve consisted of 7 time points.
[e] E4300 (pIT-SL-*pR-lacZ*)$_\lambda$ (pIT-CH-*pR-gfp*)$_{HK}$ pUHA-1.
[f] In this group, each growth curve consisted of 10 time points.
[g] In this group, each growth curve consisted of 7–10 time points.

bias the linear fitting towards a reduced slope (reduced growth rate) and hence increased doubling time.

Growth curves were measured for a number of important strains in this thesis, and each curve fit using one of the Gompertz models. The resulting doubling times are listed in Table 3.2. Given the small number of data points in the growth curves (7–10 time points as indicated in the table), growth rates were determined, with one exception[5], from fits

[5] The exception was for one of the two 96-well plate growth rate assays of the WR-MC MFL strains, where a 4 parameter Gompertz model was preferred. The Gompertz models are fit to relative OD_{600}s that

using the 3 parameter Gompertz model.

The list of basic strains in Table 3.2 includes a selection of precursor strains used to build the final MFL strains. Growth of those strains was followed in LB without added IPTG, but with different antibiotics depending on the plasmids contained in each strain. This was designed for measuring the extra burden on cell growth that was imparted by each additional plasmid and antibiotic. With such a small number of growth curves measured for each of those strains, the confidence limits are quite wide and there were no significant differences observed between these strains.

Growth curves for the induction reporter and MFL strains were obtained during the course of the hysteresis assay. As shown in Appendix A.4, the rate of growth of the MFL strains did not show any clear dependence on the concentration of IPTG or the assay conditions used. Hence, those results were aggregated for each strain and the confidence limits shown in Table 3.2 are based on the aggregated data sets. This revealed a significant difference in doubling times between the base dual reporter strain (E4300DR) and all of the induction reporter and MFL strains; presumably the faster growth of E4300DR can be accounted for by the reduced number of plasmids and antibiotics required to maintain them.

There was no significant difference in doubling time between the two induction reporter strains, and indeed, for all but one of the MFL strains, no significant difference in doubling time was observed between matching pairs of *tum*$^+$ and *tum*$^-$ strains. This validates a direct comparison between experimental and control strains. Curiously, the MFL strains with the low-copy variant of the *pR-tum* plasmid, that is, those with the SC101* origin, generally appeared to grow slower than those with the medium-copy SC101 origin. With one exception, direct comparisons between matching strains (i.e., when matching *tum*$^\pm$ and *c*I RBS strength) all show a significant difference between the SC101 and SC101* variants. This is further reflected in the fact that the doubling time of the *tum*$^+$ induction reporter strain most closely matches the doubling times of the SC101 strains. These results appear to indicate that, given the same concentration of tetracycline, the cells with a higher copy-number of the resistance gene can more effectively cope with the stress placed on cells by the antibiotic and are hence able to grow marginally quicker.

The one exception in both the cases mentioned above was due to the ER-LC *tum*$^-$ strain, which had the lowest doubling time out of all those grown with tetracycline. It is unknown why this strain behaved so differently, or whether the result is biased by a poor selection of time points. To validate or rule out this exception, additional growth curves with a larger number of time points would need to be measured for this strain.

---

are calculated by normalising to the earliest (non-zero) measurement taken after time $t = 0$. For the plate in question, that early measurement was taken later than for the other plates and resulted in a statistically significant bias in the estimation of growth rate. Thus in spite of the low number (7) of data points in that assay, the 4 parameter model was used since the additional term specifically compensates for deviation of this early time point from its normalised value of 1. The resulting growth rates and fitted growth curves for this plate were more consistent with those obtained for the other plates as shown in Appendix A.3.

In spite of the significant differences observed, it is also important to realise that for all of the strains grown in media with tetracycline, the observed standard deviation was in the range of 1.1-2.7 minutes. It is impossible to say whether the standard deviation reflects errors in fitting, in measurement or in the growth rates of the cells themselves, but in any case, it means that the estimated doubling times are highly variable and a guide to the underlying rate of cell growth at best. With this in mind and for the sake of simplicity, two common doubling times were chosen to represent the MFL strains for subsequent modelling and analysis: the mean of all of the SC101 strains, 28.2 ± 0.2 minutes (95% confidence), and the mean of all of the SC101* strains, 29.0 ± 0.2 minutes.

Recall that in assays with a 'normal' equilibration time, late log phase cultures (i.e., those used in LacZ assays) are reached 5.9 ± 0.4 hours (standard deviation) after subculturing on average, whilst with a 'short' equilibration time, late log phase is reached after approximately 4.8 hours. Using either of the MFL doubling times, this means that 12 ± 2 doublings (standard deviation) can occur before reaching late log phase for a normal equilibration time, or about 10 doublings can occur for a short equilibration time.

Note that no correlation was observed between the growth rates of given wells and the final OD_{600}s reached at the time of assay. The variation in final OD_{600}s is probably due in large part to variations in the absolute number of cells added to each well in the subculturing step for growth to log phase. Any deviation at that step would then be exponentially amplified with each cell doubling; with each additional doubling the final OD_{600}s become more and more inhomogeneous. Minimising pipetting errors in that initial subculturing step would help to improve consistency of the final OD_{600}s. This also means that concomitant with increases to equilibration times will be increases in the variance of the final OD_{600}s; given the use of the final OD_{600}s in calculating LacZ units, measurements of *pR* activity in the MFL are also implicated.

3.4.3 Growth rates in alternative media

Since the growth rate impacts parameters that are relevant to the behaviour of the MFL (see the introduction to the previous section), this raises the possibility of optimising the MFL by modifying the growth rate. A particularly important factor in setting growth rates is the choice of growth medium, and the impacts on growth rate resulting from a selection of alternative liquid media formulations were measured for the WR-LC MFL strain. The growth curves can be found in Figure 3.14 and the doubling times derived from 4-parameter Gompertz fits of that data are listed in Table 3.3. The utility of the Gompertz model over and above the log-linear model becomes particularly evident in these curves, since measurements of cell density clearly reached saturation in a number of cases. Furthermore, with a much greater number of points in each time course, use of the 4-parameter Gompertz model is more easily justified and helps to reduce fitting biases at the earliest time points.

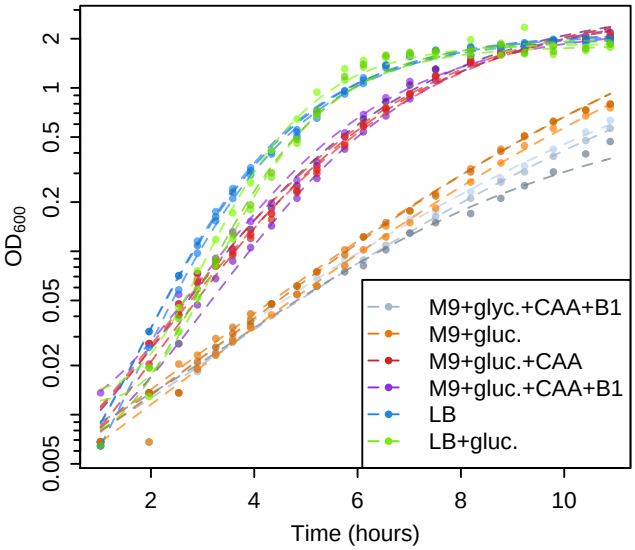


Figure 3.14: Growth curves for the MFL strain E4300DR pZC320-WR-*c*I pMTS*-*pR-tum*$^+$ grown in alternative growth media. The A_{620} was followed over time and has been converted to an OD_{600} using empirical calibration curves, which differed depending on whether the medium was based on rich (LB) or minimal (M9) broth. These media were variously supplemented with 0.2% glucose (gluc.), 0.2% glycerol (glyc.), 0.2% casamino acids (CAA), or 1 µg/mL thiamine (B1) as indicated. Growth trajectories were followed in triplicate for each medium; individual trajectories are differentiated by alternative shading. Each trajectory was fit by a 4-parameter Gompertz model (dotted lines).

Table 3.3: Growth rates determined for E4300DR pZC320-WR-*c*I pMTS*-*pR-tum*$^+$ in alternative growth media. All media were supplemented with 50 µg/mL kanamycin, 30 µg/mL ampicillin and 4 µg/mL tetracycline.

Media	**Doubling time**[a]
M9 minimal, 0.2% glycerol, 0.2% CAA[b], 1 µg/mL thiamine	83 ± 17
M9 minimal, 0.2% glucose	77 ± 1
M9 minimal, 0.2% glucose, 0.2% CAA[b]	45 ± 6
M9 minimal, 0.2% glucose, 0.2% CAA[b], 1 µg/mL thiamine	44 ± 2
LB	29 ± 3
LB, 0.2% glucose	29.4 ± 0.4

[a]Mean doubling time listed in minutes with 95% confidence limits estimated from each set of repeats ($n = 3$); each repeat was derived from a 4-parameter Gompertz fit of a 21 point time course assay.
[b]CAA: Casamino Acids.

The medium used for the measuring hysteresis in the MFL strains was based on the rich broth, LB. In spite of the large confidence limits, the mean doubling time measured in this assay matched well with the equivalent measurement listed in Table 3.2. Surprisingly, the addition of extra glucose to the broth did not further increase the rate of growth; if anything, the doubling time was *increased* over that of the original formulation. Evidently there was already an overabundance of nutrients and sugars in plain LB.

In contrast, when grown in M9 minimal medium [Miller et al., 1972, and the Materials and Methods (Chapter 8)], the doubling time for this MFL strain was significantly slowed. The most basic medium (M9 minimal with 0.2% glucose) produced especially slow growth with a doubling time of over an hour. Suspecting some metabolic deficiency for the base E4300 strain, minimal media supplemented with casamino acids (hydrolysed casein protein containing almost all the essential amino acids) and additionally the vitamin thiamine

were also trialled. The presence or absence of thiamine, initially thought to be a necessary for the E4300 strain, made no apparent difference to the rate of growth. However, supplementing the base M9 medium with amino acids resulted in a doubling time almost half that of the original. Growth in minimal media also showed sensitivity to the choice of carbon source. Changing from glucose to glycerol, even with all the additional supplements, reduced the rate of growth essentially to that of the base minimal medium without casamino acids.

3.4.4 Cell volume

The cell volume is an important factor used to convert between numbers of molecules per cell and intracellular molecular concentrations. Such conversions are highly relevant since binding constants for equilibrium reactions, such as the Tum−CI or CI−*pR* interactions, are typically derived in units of molar concentration, whilst steady-state quantitation of intracellular moieties is most easily derived in terms of a number of molecules per cell. Here formulæ are presented for estimating cell volumes from growth rates and useful conversion factors are derived. These conversion factors will be useful both for quantitation of steady-state protein levels, but also later in Chapter 6, where a stochastic model of the MFL is considered.

Following the analysis in the supplementary material of Palmer et al. [2009], the most relevant volume to consider is that of the cytoplasm (i.e. the volume of the cell minus the volume due to membranes and periplasm). The lengths, widths and volumes of *E. coli* cells can be related to the growth rate [Donachie and Robinson, 1987], with the length of K-12 given by:

$$\text{Mean cell length} = 2.0 \times 2^{(\mu/3)}\ \mu\text{m}, \tag{3.4}$$

where μ is the number of doublings per hour. Donachie and Robinson [1987] report that the same cells have a length to width ratio of around 4.9, so for a doubling time of 28.2 minutes, the average cell length would be ∼3.3 μm with a width (diameter) of ∼0.67 μm. The periplasmic space[6] has a width of around 13 nm for K-12 *E. coli* [Graham et al., 1991], the outer membrane a width of around 13 nm [Bayer, 1991], and the inner membrane a slightly smaller width (set to 8 nm here). Hence, the width of the shell encapsulating the cytoplasmic volume sums to ∼34 nm, implying that the cytoplasm (using the dimensions listed above) would have a length of ∼3.2 μm with radius ∼0.30 μm. Assuming a cylindrical volume, this leaves a cytoplasmic volume of 0.90 μm^3. Using that cytoplasmic volume and Avogadro's constant, the effective concentration of one molecule per cell can be calculated as 1.8 nM.

[6]The periplasmic space is the space between inner and outer membranes in gram-negative bacteria.

3.5 Balancing the MFL module

Rates of production and loss of regulatory proteins are of fundamental importance in modelling the dynamics of genetic circuits. Determining the nature and rate of each of these two processes for Tum and CI in the MFL will provide a grounding first step into understanding the behaviour of this synthetic switch. This section starts with a description of the techniques used to quantify specific proteins of interest *in vivo*, and then moves on to consider the degradation rates of Tum and CI. The estimates for rates of loss are then used to estimate promoter production rates from *in vivo* steady-state measurements of Tum and CI concentrations.

3.5.1 Quantitating intracellular proteins

A measure of *in vivo* protein concentrations for specific targets is necessary for determining both degradation and production rates of that target. Here we use Western blots of cell extracts in combination with Tum- or CI-specific antibodies to quantify each of the proteins by comparison with known concentrations of purified protein.

Purified stocks of both $TumHis_6$ [Shearwin et al., 1998] and CI [Shearwin and Egan, 1996] have been obtained and quantified previously, and rabbit polyclonal antibodies have been raised against both of these stocks. For Western blotting, whole cell extracts of defined OD_{600} and volume are separated by sodium dodecyl sulphate polyacrylamide gel electrophoresis (SDS-PAGE) and transferred to polyvinylidene fluoride (PVDF) membranes; the primary α-$TumHis_6$ or α-CI antibodies and fluorescently tagged secondary antibodies then allow relative quantification of $TumHis_6$ or CI respectively from the intensities of the protein-specific bands. Further details on the α-$TumHis_6$ and α-CI antibodies and detailed protocols for preparation of cell extracts and for Western blotting can be found in the Materials and Methods (Chapter 8).

By including reference concentrations of $TumHis_6$ or CI on each blot using the purified stocks, per-blot calibration curves can be derived; an example of a $TumHis_6$ calibration curve is shown in Figure 3.15. The response of band intensity to mass of protein loaded in the lane is sigmoidal, indicating that sensitivity is lowered at the extrema. The curve is well-fit by a three parameter log-logistic (i.e., Hill) curve as shown. Background noise reduces sensitivity at low concentrations, whilst reduced capacity for transfer at high Tum concentrations is the most likely cause for loss of sensitivity at the highest intensities. As will be seen in Sections 3.5.3 and 3.5.4, the protein gels used to preparing the Western blots have limited lane space, reducing the number of calibration samples per blot to four. As such, the basal parameter in the four parameter log-logistic fit is fixed to zero to reduce the number of regression parameters.

Ideally such calibration samples would be included on each Western blot, and a calibration curve calculated to represent each blot individually. This would account for blot-to-

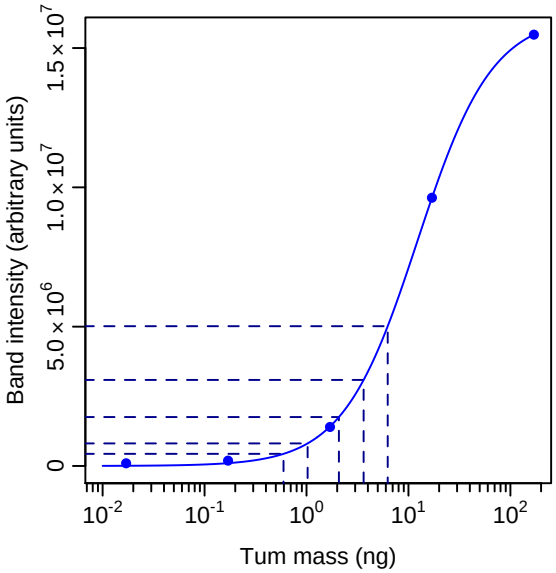


Figure 3.15: Fitting the sigmoidal response of band intensity to TumHis$_6$ mass. Band intensities measured for Western blots of known quantities of TumHis$_6$ reveal a sigmoidal response on TumHis$_6$ mass that is well fit by a four parameter log-logistic function with basal parameter fixed to zero. TumHis$_6$ mass is shown on a log-axis for easier visualisation of the smaller masses. The dashed lines illustrate quantitation of TumHis$_6$ from a base two dilution series of an *in vivo* extract of a strain with the *tum*4$^-$5$^-$*his* gene; given the log scaling, equal spacing between the serial dilutions is anticipated.

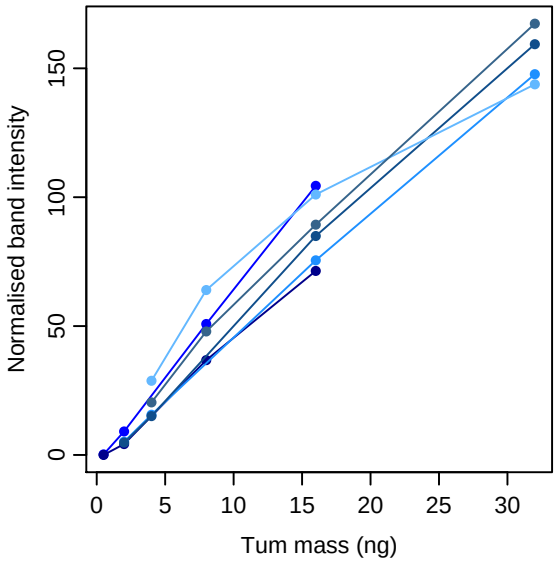


Figure 3.16: Using Box-Cox transformation of Western blot band intensities to derive a quantity with linear dependence on Tum mass. The TumHis$_6$ calibration curves taken from Western blots in Section 3.5.3 were first normalised against the minimum intensity band for a more convenient scale, subject to Box-Cox transformation with $\lambda = 1.5$ and then plotted versus the known Tum mass.

blot variation resulting, for example, from differences in transfer efficiency and incubation times with primary and secondary antibodies. However, it was not always possible to fit such extracts on all blots, so a more generic transformation of band intensities was sought. The Box-Cox transformation introduced in Section 3.3 can again be applied here to find a transformation of all the calibration data sets that maximises the likelihood of a linear fit of the data. When such a transformation is applied to raw band intensities, it should produce a quantity that has a linear dependence on the mass of protein present in that band. This was needed for the TumHis$_6$ blots only, so for all of the available TumHis$_6$ calibration data (shown in Section 3.5.3) the log-likelihood of a linear model incorporating Box-Cox transformation of the data was profiled to give an optimum $\lambda = 1.5$. The application of this transformation to the calibration curves is shown in Figure 3.16 to demonstrate the (limited) efficacy of this linearisation scheme.

In summary, for quantitation of TumHis$_6$ levels by Western blot, Hill curve fits of calibration samples will be used to calibrate blots where available, otherwise the raw band intensities will be normalised using the Box-Cox method.

3.5.2 Degradation rates of Tum and CI

The *in vivo* half-lives of regulatory proteins are essential for determining the dynamics of gene networks. At steady-state, the rate of loss of a protein, either by dilution or active degradation, balances the rate of production and together these rates determine steady-state concentrations. Indeed, proteolysis plays important roles in gene regulation with significant numbers of key regulatory genes having very short half-lives [Gottesman and Maurizi, 1992]. These fast degradation rates enable quicker responses to changes in production rate. In contrast, slow degradation rates result in increased time-averaging of the input signal, where the undegraded protein essentially acts as a buffer for fluctuations in the production rate.

In *E. coli*, most proteins are quite stable, with the dominant mechanism for protein loss occuring through the dilution that results from cell division; the growth rate is an important measure of half-life for such proteins. We have previously observed that the CI protein of phage 186 is stable *in vivo* [Keith Shearwin (KS), unpublished data], but had little *a priori* reason to suspect that Tum would be actively degraded. Nonetheless, for completeness the half-lives of both Tum and CI in strains resembling those of the MFL were determined.

To enable optimisation of Tum concentration for the degradation assay, a strain with an inducible level of Tum was chosen. The Tum activity assay strain, E4300 (λRS45 *pR-lacZΔYA*) (pAH144-P_{bla}-*c*I)$_{\text{HK}}$ pUHA-1 pZE15-*tum*4^{-}5^{-}*his*, can be used to report on Tum relief of CI repression of *pR* by LacZ assay, and such assays along with further strain details can be found in Chapter 9. However, for the purposes here, the Tum activity strain is merely a convenient (and decoupled) source of both Tum and CI. Importantly, unlike the MFL strains, it does not have any chloramphenicol resistance, which is likely to bias the assay itself given the use of chloramphenicol to halt protein production (discussed below). The strain also has a source of CI, which is produced constitutively from the medium-strength P_{bla} promoter in single copy. Whilst the level of CI thus produced would be low, it should still be possible to observe the putative absence of degradation.

Degradation of CI or Tum may be influenced by the presence or absence of the binding partner. As such, sibling strains without a source of CI were cloned with the integrated but empty (pAH144)$_{\text{HK}}$ module replacing (pAH144-P_{bla}-*c*I)$_{\text{HK}}$, and *tum*$^{-}$ strains were prepared by using the empty pZE15 plasmid instead of pZE15-*tum*4^{-}5^{-}*his*. This produced four alternative assay strains with all combinations of *tum*$^{\pm}$ and *c*I$^{\pm}$. In order to maximise comparability with the MFL strains, the pMTS-*pR-tum*$^{-}$ plasmid was additionally transformed into each of the assay strains. This meant that the antibiotics used during growth could be the same as in the MFL strains, with the exception that a higher concentration of ampicillin was required (Ap^{100}) due to the high copy origin (ColE1) of the pZE15 backbone.

To measure protein half-lives, protein production in log-phase cultures was stopped

by the addition of chloramphenicol, and a time course series of extracts then prepared from these cultures. In theory, given the absence of protein production and the consequent halting of cell growth, changes to protein concentration should only occur as a result of the existing host degradation machinery. Chloramphenicol is an antibiotic that targets the 50S ribosomal subunit of elongating ribosomes in *E. coli* and is considered to be bacteriostatic. Unfortunately, it does in fact decrease cell viability and cause limited cell lysis within about 30 minutes. However, with the addition of 50 mM $MgSO_4$, cell wall integrity is seemingly stabilised and this effect is abated [Gupta, 1975]. It has also been observed that addition of just $MgSO_4$ to the medium slows growth [David Priest, unpublished data], and this would only serve to aid the bacteriostatic effect. Given the time-sensitive nature of the assay and that protein half-lives can be on the order of minutes, culture samples were first aliquoted into much larger volumes of ice-cold Phosphate-Buffered Saline (PBS) to quickly 'freeze' the cell states prior to extract preparation. Further details on the preparation of degradation extracts can be found in the Materials and Methods (Chapter 8).

Western blots of such extracts are presented in Figure 3.17 for two different inducer concentrations. These blots demonstrate that $TumHis_6$ is indeed degraded in E4300, but the extent of degradation is vastly reduced in the presence of even the small amount of CI produced from single-copy P_{bla}. Furthermore, loss of Tum appears to have stopped by about 20 minutes, but relative to the equivalent Tum^- extracts, a fixed level of Tum protein appears to persist for at least 60 minutes.

α-Tum blots of the degradation extracts in Figure 3.17, and equivalent α-CI blots of time-course extracts for both of the CI^+ strains (data not shown), were quantitated as per Section 3.5.1 and the Materials and Methods (Chapter 8). The resulting time course curves are shown plotted in Figure 3.18 on the next page. There was little to no active degradation of CI in either of the CI^+ strains over the course of the assay, as can be seen in Figure 3.18(a). Thus, as anticipated, CI appears stable for at least a typical generation time. This confirms

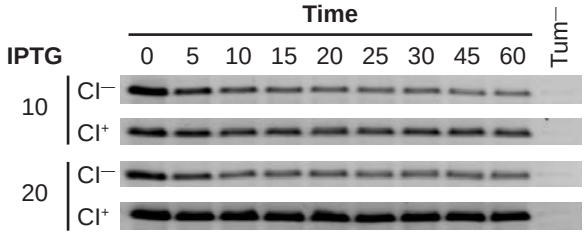


Figure 3.17: Tum is degradaded as observed by Western blot with an α-Tum antibody[a]. Protein production in cultures of E4300 (λRS45 *pR-lacZ*Δ*YA*) (pAH144-P_{bla}-*cI*)$_{HK}$ pUHA-1 pMTS-*pR-tum*$^-$ pZE15-*tum*4$^-$5$^-$*his*, with either 10 or 20 µM IPTG, were stopped at time 0 by addition of chloramphenicol and samples taken over the following hour. CI^- strains have (pAH144)$_{HK}$ instead of (pAH144-P_{bla}-*cI*)$_{HK}$ and Tum^- strains (sampled only at time 0) contain the empty pZE15 plasmid.

[a]Note that for optimised printing, the grayscale levels of all of the blot images in this and the following figures have been adjusted; analysis is always performed on the unaltered images, however.

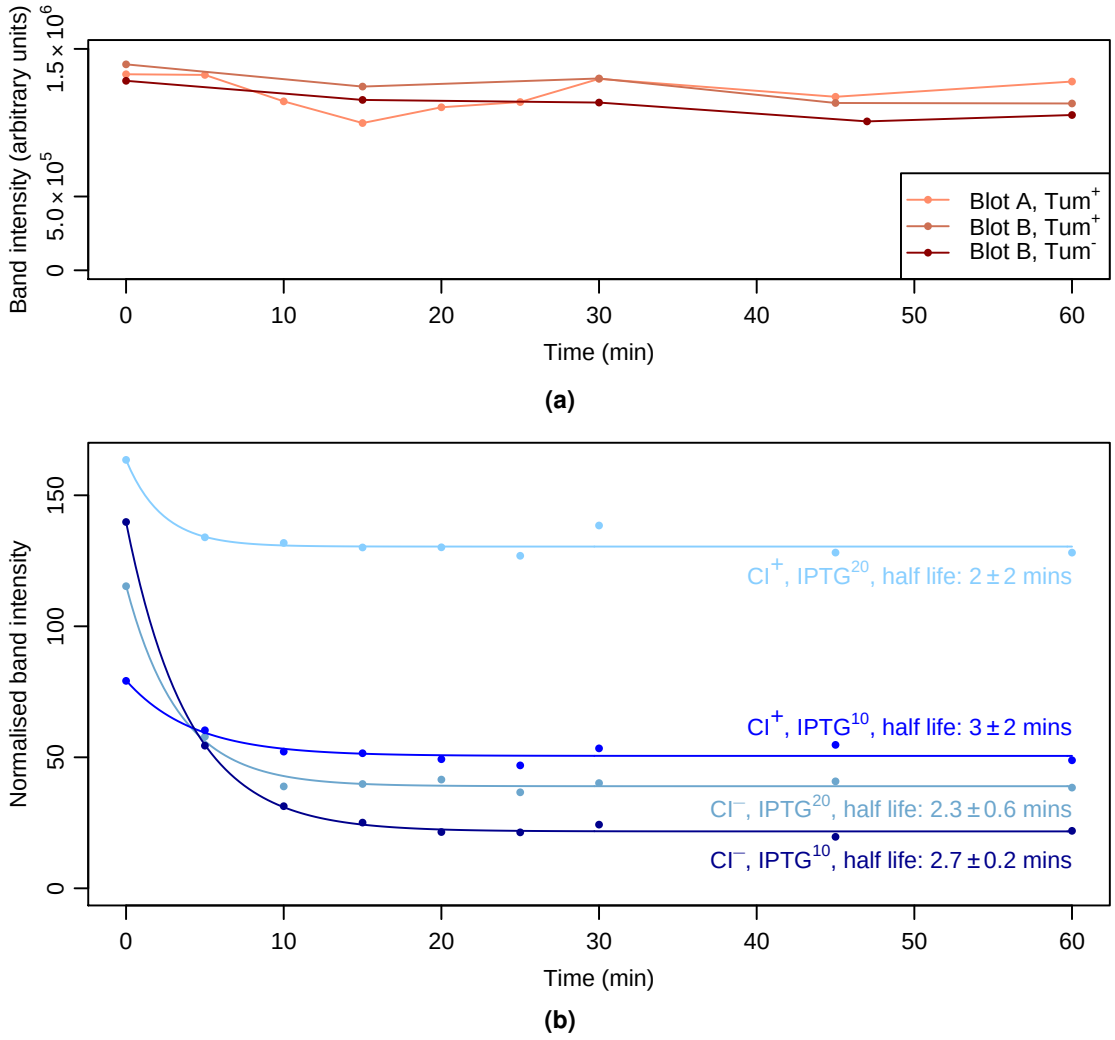


Figure 3.18: Measuring degradation of Tum and CI in E4300 (λRS45 *pR-lacZΔYA*) (pAH144-P_{bla}-*cI*)$_{HK}$ pUHA-1 pMTS-*pR-tum*$^{-}$ pZE15-*tum*4^{-}5^{-}*his*. **(a)** Within a typical cell generation time, there is no evidence for active degradation of CI whether in the presence of Tum or not. Plotted are the quantitated CI-specific bands of degradation extracts of cultures grown with 20 µM IPTG. In the Tum^{-} strain, pZE15-*tum*4^{-}5^{-}*his* is substituted with empty pZE15. **(b)** Measuring Tum degradation for cultures grown with 10 or 20 µM IPTG. Plotted are the quantitated Tum-specific bands of Figure 3.17, which were additionally normalised using the Box-Cox transformation (see Figure 3.16), and fit using a non-linear model of exponential decay using a offset term to account for the non-zero asymptote. Half-lives from the fits are indicated with 95% confidence limits. In the CI^{-} strain (pAH144-P_{bla}-*cI*)$_{HK}$ is substituted with (pAH144)$_{HK}$.

that the dominant mechanism for loss of CI is via the dilution that results from cell growth, and the parameter for loss of CI is best calculated from the growth rate as described in Section 3.4.2. Given that this result is the same whether Tum is present or not, it is clear that inactivation of CI by Tum does not occur by targetting CI for degradation.

Whilst Tum is certainly degraded, the degradation appears to be limited to a subset of the total Tum mass. Possible reasons for the unexpected non-zero asymptote will be discussed below, but by including an offset term in the fitting procedure, estimates for Tum half lives in the different strains can still be made. Due to the additional offset, non-

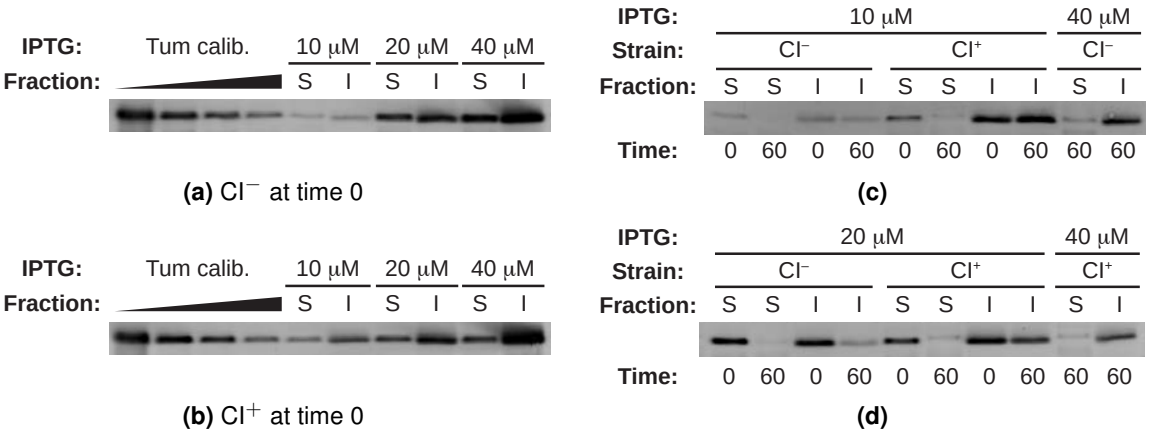


Figure 3.19: Comparing soluble (S) and insoluble (I) fractions of Tum at initial and final time points of the degradation assays. Samples of E4300 (λRS45 *pR-lacZΔYA*) $(P_{\text{bla}}\text{-}cI)_{\text{HK}}$ pUHA-1 pMTS-*pR-tum*$^-$ pZE15-*tum*4$^-$5$^-$*his* (CI$^+$; CI$^-$ strains have $(\text{pAH144})_{\text{HK}}$ instead of $(P_{\text{bla}}\text{-}cI)_{\text{HK}}$) were analysed by Western blot and further divided by the concentration of IPTG inducer used during growth to log phase. Extracts were prepared from cultures with an OD_{600} around 0.6 and the equivalent of 120 μL of each culture was loaded per well. **(a)** and **(b)** Comparison of insoluble and soluble fractions at the initial time point in the CI$^-$ and CI$^+$ strains. The wedges indicate increasing dilutions of a Tum calibration (calib.) stock made up in Tum$^-$ cell extracts (purified TumHis_6 at 32 ng/12 μL was serially diluted to 1, 2, 8 and 32 times). **(c)** and **(d)** Direct comparison of insoluble and soluble Tum fractions at initial and final time points.

linear regression of the intensity (I) against time (t) was performed using the following model of exponential decay:

$$I = Ae^{\frac{\log(2)}{h}t} + B, \tag{3.5}$$

where B is the offset, $A + B$ is the intensity at $t = 0$ and h is the half life. The half-life is consistent across all the assays; degradation seems especially quick with half lives of around 2–3 minutes. Given the low number of samples taken in the time interval of greatest change (around 0–10 minutes), some of the estimates have low confidence, especially in the CI$^+$ strains where the amount of Tum available for degradation seems reduced relative to the CI$^-$ strains. The lower intensity seen at time zero for the degradation extract at 20 μM versus 10 μM IPTG induction, is likely a result of some small delay in taking the sample after addition of chloramphenicol; given the short half-life, even small delays could mean large differences in intensity. Further work needs to be done to take additional samples at the early time points so that confidence in the shapes of the degradation curves and derived rates can be increased.

The fraction of Tum still left at the end of the time courses could conceivably be explained either by short half lives of the proteases involved in Tum degradation, or by a subset of Tum that is protease-inaccesible. In attempts at purifying Tum, it was found that over-expression causes some fraction of Tum to become insoluble. Thus to assess whether this insoluble fraction may somehow be protected from degradation, trial degradation assays were run as before, except that samples were only collected at the $t = 0'$ and $t = 60'$ time points and cell extracts from those samples were split into insoluble and soluble frac-

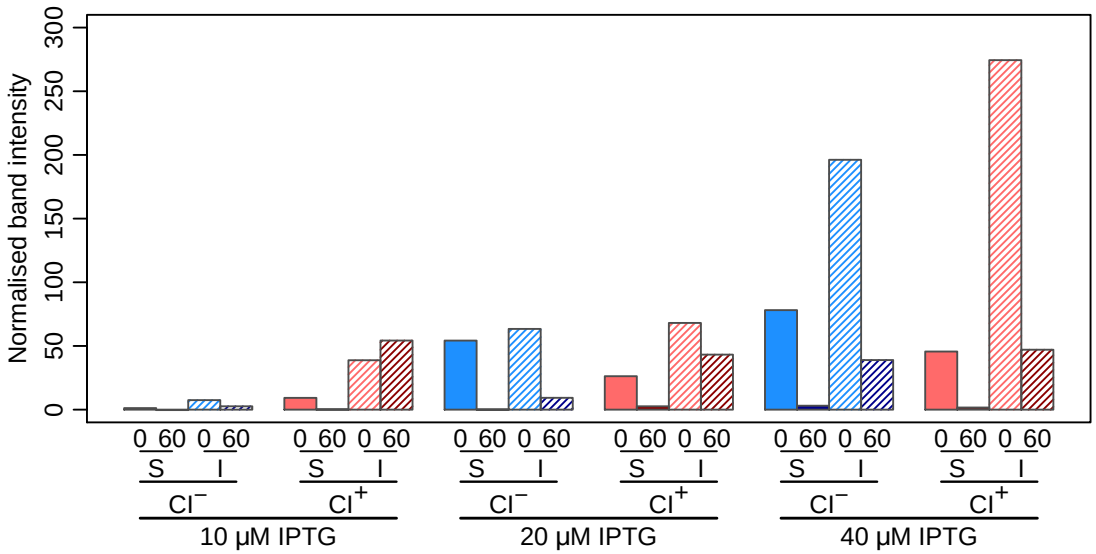


Figure 3.20: The degradation-resistant fraction of Tum appears to be a subset of the insoluble fraction. Cultures of E4300 (λRS45 *pR-lacZΔYA*) $(P_{bla}$-*cI*$)_{HK}$ pUHA-1 pMTS-*pR-tum*$^-$ pZE15-*tum*4^-5^- *his* were grown with induction at 10, 20 or 40 μM IPTG to log phase. Protein production was stopped by the addition of chloramphenicol and cell extracts of samples taken at 0 and 60 minutes after addition of chloramphenicol were split into soluble (S) and insoluble (I) fractions and quantitated by Western blot. Shown are the Tum-specific band intensities after Box-Cox transformation, as described in Figure 3.16, and normalisation by the OD_{600}s measured just prior to addition of $MgSO_4$ (i.e., 10 minutes before addition of chloramphenicol). The CI^- strains had $(pAH144)_{HK}$ instead of $(P_{bla}$-*cI*$)_{HK}$.

tions. To also gauge the levels of Tum present at each of the different levels of induction, calibration extracts were also prepared by adding known concentrations of purified $TumHis_6$ to extracts prepared from Tum^- cultures (E4300DR pZC320 pMTS-*pR-tum*$^-$). Western blots of these extracts are shown in Figure 3.19. Like for the previous observations made from Tum expression vectors, a substantial portion of Tum was observed to lie in the insoluble fraction even at low induction levels. From the side-by-side comparisons in Figures 3.19(c) and 3.19(d), in particular, it can also be seen that, whilst almost all of the soluble fraction appears lost by $t = 60$, the levels of Tum in each of the insoluble fractions at the same time point are still clearly visible.

For a more objective analysis, the band intensities from all of the blots in Figure 3.19 were quantified and the Box-Cox transformation described in Section 3.5.1 applied to increase comparability. These estimates were also normalised using the OD_{600} measurement made just prior[7] to addition of $MgSO_4$ to account for differences in the densities of cells loaded into each lane. The resulting normalised intensities are depicted in Figure 3.20 to reveal that the soluble fractions do indeed all start with non-zero levels of Tum at time $t = 0$, but which are all then reduced close to zero by time $t = 60'$. In contrast, whilst substantially reduced, the insoluble fractions are all still clearly non-zero by $t = 60'$. This would

[7]It was noted that the addition of $MgSO_4$ appeared to reduce the OD_{600} more than would be anticipated simply by dilution due to the additional volume. As such, the prior OD_{600} was assumed to better compare with OD_{600}s measured elsewhere in spite of the additional growth that may have occurred during the 10 minute incubation.

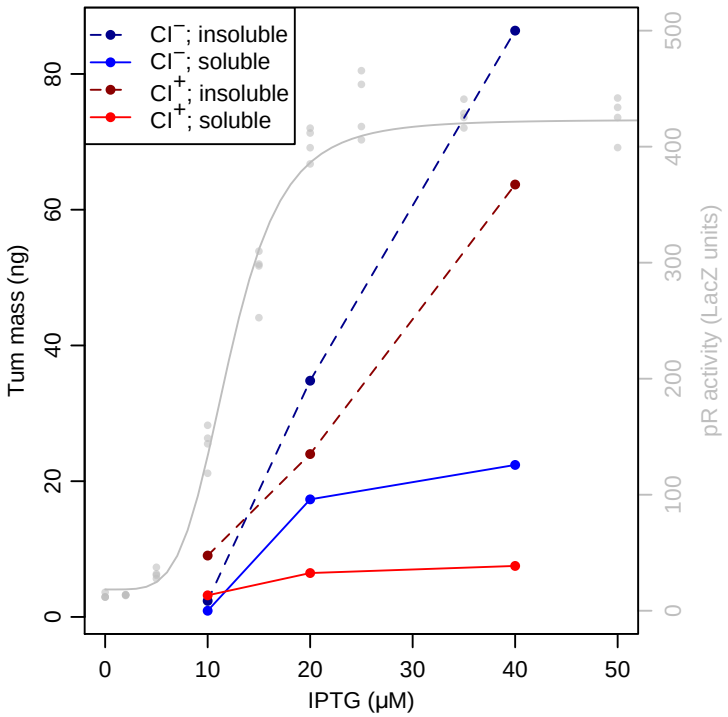


Figure 3.21: The fraction of soluble Tum present at steady state appears to saturate as the production rate is increased. Time $t = 0$ extracts from degradation assays of strains including an IPTG-inducible level of Tum were split into soluble and insoluble fractions and probed by Western blot alongside known quantities of purified Tum (see Figures 3.19(a) and 3.19(b)). The masses of Tum in each extract could then be calculated via fits of these calibration samples as described in Section 3.5.1. The mass estimates were normalised to the mass expected for 135 µL of culture at OD_{600} 0.6 by using the OD_{600}s measured just prior to addition of $MgSO_4$ in the degradation assays. For comparison, plotted in the background in grey are the results of a LacZ assay measuring *pR* activity of the CI^+ strain (as described in Figure 3.19 but *without* pMTS-*pR-tum*$^-$); in that strain, *pR* activity acts as a proxy to report on Tum antirepressor activity.

suggest that at least a portion of the insoluble fraction of Tum is protease-inaccessible. Further, if the insoluble fraction is in slow equilibrium with the soluble fraction, the entire insoluble fraction could be considered degradation resistant; as soluble Tum is degradaded, the equilibrium may shift to release insoluble Tum into the soluble fraction.

Interestingly, the proportion of Tum in the insoluble fraction relative to that in the soluble fraction appeared to increase with the level of induction. This is made more apparent in Figure 3.21 where the extracts blotted alongside calibration samples are shown quantified as mass estimates. These estimates were also normalised using the OD_{600} measurement made just prior to addition of $MgSO_4$, but for comparison with subsequent measurements of steady-state Tum concentrations (Section 3.5.3), the normalisation was also designed to represent the mass in 135 µL of culture with OD_{600} 0.6 (note that in the present samples, the equivalent of 120 µL of culture was loaded per well). Only the $t = 0$ samples were analysed in this way, since these approximate the distribution between soluble and insoluble fractions at steady state. With such a limited data set it is hard to draw conclusions about the two fractions, and certainly more experiments will be required to understand the roles of each Tum fraction and their impact on Tum activity. However, the trends in the data do seem to indicate that as the production rate of Tum is increased, the level of Tum in the insoluble fraction continues to increase, whilst the level of Tum in the soluble

fraction reaches a saturation point. Curiously, this saturation point appears to correspond with the saturation in Tum activity observed in LacZ assays of the original Tum activity assay strain (i.e., without the pMTS-*pR*-*tum*$^-$ plasmid). Here this is tentatively taken as an indication that the soluble fraction is the active fraction.

From the degradation curves in Figure 3.18, it is clear that a fraction of Tum is degraded with a very short half life (around 2–3 minutes), with a degradation-resistant fraction that is lost by cell division. Then from the analysis of the soluble and insoluble fractions in Figure 3.20, it also seems that the degradation-resistant fraction is at least a subset of the insoluble fraction. Furthermore, the trends of each fraction shown in Figure 3.21 suggest that the equilibrium between the two fractions reaches a saturation point in terms of the soluble fraction. To add to the complexity, it can also be seen that on the whole, the CI$^-$ strains seem to have more soluble Tum than the CI$^+$ strains do at $t = 0'$, though, at this point it is too difficult to say what role CI may play in disrupting the balance between the two fractions. Taken together these results present a rather complicated model for the state of Tum, its degradation and its activity. So to keep the model as simple as possible given the limited data, it is presumed that only the soluble fraction of Tum is active and that this fraction is produced from *pR* at a rate that is directly proportional to the repression of *pR*. This means that only the evolution of soluble Tum need be considered, with a degradation rate as determined from the degradation curves and a production rate chosen to match the steady-state level of *soluble* Tum. This would clearly be a poor approximation for large production rates, but as will be seen later, the unrepressed *pR* promoter produces Tum at a rate less than that seen here for production from P_{lac} induced using 20 μM IPTG. For production rates below 20 μM saturation of the soluble fraction does not appear to have been reached, so the linear approximation as described may well be good enough for modelling the MFL.

3.5.3 Production rate from *pR*

Protein production in *E. coli* involves a complex series of steps, from binding of the closed RNAP complex at the promoter site and isomerisation to its transcriptionally active open complex, through transcription of the operon into mRNA and subsequent translation by bound ribosomes, to folding of the nascent peptide into an active protein. Matters are further complicated by a cohort of regulatory mechanisms that have been found to occur at each step along the pathway. In spite of this complexity, simple models for the entire production process can still capture behaviours that are relevant on the time-scales of gene-networks [Sneppen et al., 2010].

For the phage λ switch, the rate-limiting step in the production of repressor and Cro proteins was found to be the transition of RNAP from the closed to the open complex [Shea and Ackers, 1985]. This motivated a one-step model of production, where the protein production rate was simply the rate of isomerisation weighted by the statistical thermo-

dynamic occupancy of the promoter by RNAP. Elsewhere, models of gene networks often employ a two-step description of protein production, where the mRNA transcript becomes an intermediate in the model [Hlavacek and Savageau, 1996]. This additional level of complexity is primarily used for timing-sensitive circuits such as oscillators [Elowitz and Leibler, 2000], or in stochastic models that account for the additional noise generated at this step [McAdams and Arkin, 1997]. The MFL model employed in François and Hakim [2005] emphasised the importance of the mRNA intermediate in the dynamics of an *oscillating* MFL.

Here mRNA dynamics are ignored and the one-step model is employed, which gives a rate equation for the evolution of protein A in the absence of transcriptional regulation as $\frac{\mathrm{d}A}{\mathrm{d}t} = p + \delta A$, where p is the representative rate of production, and δ is the rate of degradation of the protein. Note that the short Tum half-life (2–3 minutes) found in the previous section could be cause for concern given that the mean mRNA half-life in *E. coli* is 5.2 minutes [Bernstein et al., 2002]. However, in the absence of any data regarding specific *in vivo* mRNA half-lives for any of the MFL transcripts, mRNA dynamics are ignored regardless. This assumption may be justified in part by the anticipated steady-state behaviour of the bistable circuit; the primary error in making the assumption would be an overestimate of the time taken for Tum to reach steady state. This has no direct consequence for a purely steady-state description, and given that loss of CI and the two reporters (GFP and β-gal) is dilution limited with half-lives on the order of the cell doubling time (much longer than 5.2 minutes), the evolution of those products remains the primary delay for reaching steady state.

At steady state, production rates are exactly balanced by degradation rates: in the one-step model, $\frac{\mathrm{d}A}{\mathrm{d}t} = 0$, so $A = p/\delta$. Hence, by using the degration rates determined in the previous section, production rates for the two MFL promoters, pR and P_{lac}, can be determined by measuring the steady-state levels of their gene products, Tum and CI respectively. To best inform the modelling some care should be taken to ensure that measurements are indeed made at steady state, and that the states of each promoter are well defined and unchanging. For these reasons, the MFL interactions are decoupled by making minor simplifications of the strains before taking steady-state measurements.

In the MFL, the rate of production from pR is further weighted by the proportion of CI bound at the promoter, but the extent of this repression can easily be described by a Hill curve that scales production relative to some maximum rate as described mathematically in Section 2.2.1. As a result, here the activity of unrepressed pR is calculated by measuring the steady-state TumHis_6 levels that are produced in an MFL-equivalent strain without a source of CI. This parameter is measured only for the pMTS-*pR*-*tum*$^+$ plasmid with an SC101 origin, and not for the low-copy version with an SC101* origin, but by scaling according to the difference in copy numbers a value for the low-copy plasmid can be derived.

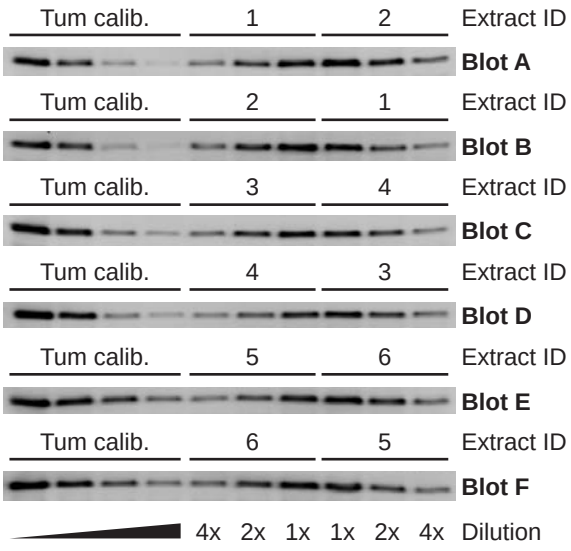


Figure 3.22: Western blots for quantitating steady-state production of Tum from *pR* in E4300DR pZC320 pMTS-*pR*-*tum*$^+$. The wedge indicates increasing dilution factors for the calibration samples; these were all prepared from master stocks of either 16 ng/12 µL (blots A and B) or 32 ng/12 µL (blots C–F) of purified TumHis$_6$ made up in Tum$^-$ cell extracts. The (serial) dilutions made from these calibration samples included 1, 2, and 8 times for all blots, but added 32 times dilution for blots A and B, 16 times for blots C and D, and 4 times for blots E and F.

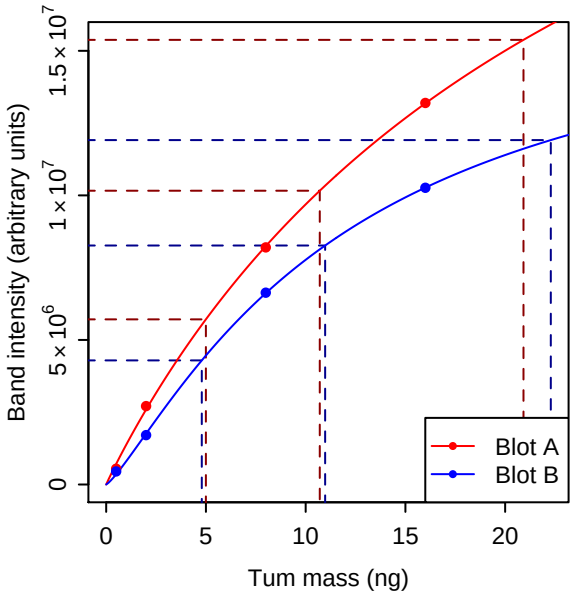


Figure 3.23: Calibrating Tum-specific band intensities on Western blots with Tum mass. The data shown is derived from blots A and B in Figure 3.22, and the solid lines are four-parameter log-logistic fits of the calibration data (with the basal parameter fixed at zero). The dotted lines indicate the Tum-specific band intensities of the dilution series of extract 2 and the corresponding estimates of mass as determined by the calibration fit.

Quantitative extracts of log-phase cultures grown from six independent colonies of E4300DR pZC320 pMTS-*pR*-*tum*$^+$ were prepared for subsequent analysis. Since Tum protein production and the state of the *pR* promoter are decoupled from the MFL circuit in this strain, there was no reason to anticipate a lengthy time to reach steady state from the stationary-phase O/N cultures. As such, O/N cultures were diluted for growth to log phase according to the short equilibration time protocol. Once grown to an OD_{600} of approximately 0.6, the volumes used for preparing the extracts were normalised so that each sample contained the equivalent of 900 µL of culture with OD_{600} 0.6. For preparing the calibration extracts and diluting the *tum*$^+$ extracts, an extract of the equivalent pMTS-*pR*-*tum*$^-$ strain was also prepared using the large scale protocol; see the Materials and Methods (Chapter 8) for further technical details.

Western blots of the resultant cell extracts probed with the α-Tum antibody are shown in Figure 3.22. As alluded to earlier, four of the ten lanes were reserved for a dilution series of the calibration sample (purified and quantified TumHis$_6$ made up in the Tum$^-$ extract), with the other six lanes reserved for a dilution series of the Tum$^+$ experimental extracts (also diluting in Tum$^-$ extract). As shown in Figure 3.23, calibration curves were

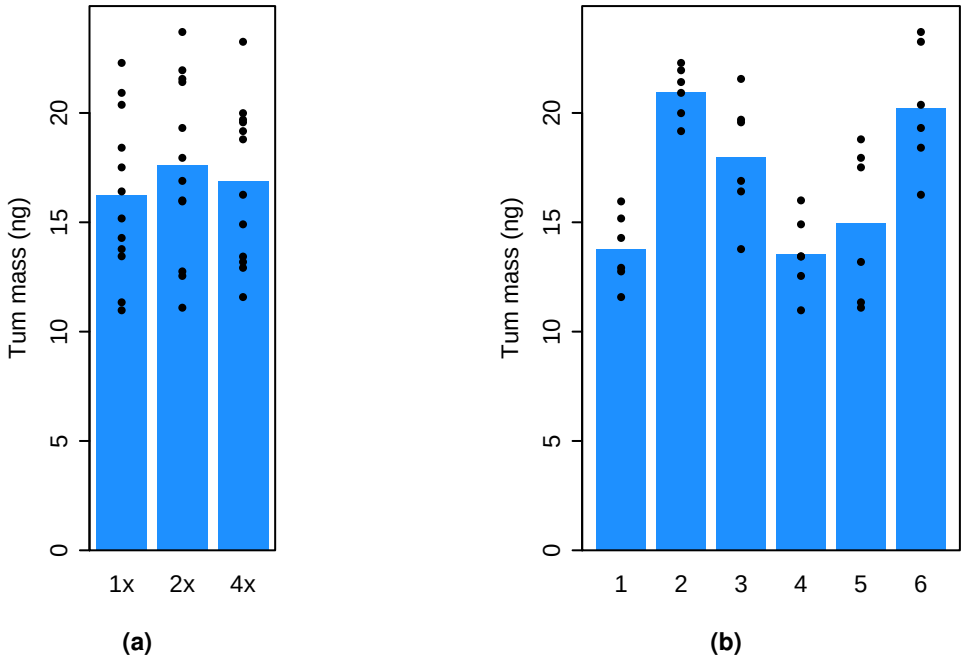


Figure 3.24: Quantifying Tum mass in an MFL-equivalent strain with the pMTS-*pR-tum*$^+$ plasmid but without the *c*I gene. Plotted are estimates of Tum mass in 135 µL samples of log-phase cultures with OD_{600} 0.6. The points are individual estimates taken from each sample band in the blots in Figure 3.22. **(a)** Whilst the estimates are highly variable, the means of these estimates are independent of the dilution factor used, giving greater confidence in the form of the calibration curve. **(b)** Mean estimates of Tum mass grouped by extract ID. These means are then averaged to give the final estimate listed in Table 3.4.

fit for each blot and the mass of each extract quantified. After scaling the estimates to their undiluted values, they then represent the mass of $TumHis_6$ in 135 µL of log-phase culture[8] with OD_{600} 0.6. Bar graphs summarising these scaled estimates are presented in Figure 3.24.

The non-linear form of the calibration curves appears to be a good choice judging by a few cross-checks. The curves generally match up the estimated masses of identical samples run on different blots very well (Figure 3.23). Further, similar mean values are observed for estimates of Tum mass when the estimates are split according to their dilution factor (Figure 3.24(a)). The main source of variation thus appears to originate from that occurring between the extracts themselves (Figure 3.24(b)), and for this reason, the combined estimate of Tum mass with 95% confidence limits (listed in Table 3.4 on the next page) was derived from these correlated means.

To generalise the mass estimate to a value per cell, the concentration of cells in log-phase cultures of E4300DR pZC320 pMTS-*pR-tum*$^+$ was quantified. This was performed by counting colonies on plates spread with dilution series (chosen to obtain approximately 20, 100 and 200 colonies per plate) of triplicate log-phase cultures all normalised to

[8]As per the Materials and Methods (Chapter 8), the 900 µL culture samples are resuspended in 80 µL of lysis solution/loading buffer, and 12 µL of that preparation is loaded per lane, resulting in the volume as reported.

Table 3.4: Tabulating the strengths of the *pR* and P_{lac} promoters in the MFL strains as calculated from steady-state measurements of the gene products (TumHis_6 and CI respectively). Measurement of *pR* activity was for expression from the medium-copy pMTS-*pR*-*tum*$^+$ plasmid in a strain matching the MFL strains but without the *c*I gene; measurement of P_{lac} activity was for expression from the single-copy plasmids pZC320-WR-*c*I or pZC320-ER-*c*I in strains matching the MFL but without the *tum* gene, and induced using 300 µM IPTG. For generality, the measurements are converted to molecules (mol.) per cell. Reported errors are 95% confidence limits on the mean.

Promoter	RBS[a]	Steady-state mass (ng)[b]	Cell count (c.f.u./mL)[c]	Mol. per cell
pR	–	17 ± 3	$(1.7 \pm 0.2) \times 10^8$	$(2.5 \pm 0.7) \times 10^4$
P_{lac}	W	13 ± 3	$(2.2 \pm 0.3) \times 10^8$	$(1.3 \pm 0.5) \times 10^4$
	E	33 ± 10	$(2.0 \pm 0.3) \times 10^8$	$(3.5 \pm 1.6) \times 10^4$
Promoter	**RBS**	**Half-life (min)**	**Production rate (mol./min)[d]**	
pR	–	2.7 ± 0.2	$(2.1 \pm 0.8) \times 10^3$	
P_{lac}	W	28.2 ± 0.2	$(3.3 \pm 1.2) \times 10^2$	
	E	28.2 ± 0.2	$(8.6 \pm 3.9) \times 10^2$	

[a]Ribosome binding sites differed for the P_{lac} measurements only and are either wild-type (W) or enhanced (E).

[b]The steady-state mass was measured for 135 µL of log-phase culture with OD_{600} 0.6.

[c]Counts of colony forming units (c.f.u.) are derived from colony counts of diluted cultures calibrated to be equivalent to 1 mL of culture with OD_{600} 0.6.

[d]The estimate for *pR* production rate assumes that only 33% of the steady-state Tum is in the soluble (presumed active) fraction.

OD_{600} 0.6. The colony forming units (c.f.u.) were counted and linear regression of these counts versus the relative concentration factors was used to estimate the concentration of cells in the undiluted cultures. The resulting estimate of cell concentration is again listed in Table 3.4. This concentration of cells was used to normalise the estimated mass of Tum to a measurement per cell. By using the known molecular weight of TumHis_6 (17958 Da), the steady-state concentration of Tum produced from pMTS-*pR*-*tum*$^+$ can be calculated in molecules per cell, and this value is also listed in Table 3.4. A steady-state concentration of 25000 proteins per cell is well within the range of steady-state levels observed in a large-scale assessment of absolute protein concentrations in *E. coli* [Lu et al., 2007].

In theory, the production rate of Tum from *pR* could be estimated directly from its steady-state concentration using $p = A \log(2)/h$, where h is the half-life and A the concentration of Tum. However, given the observed presence of both degradation-resistant and degradation-sensitive fractions of Tum, the model of Tum evolution needs to be treated more carefully. The simple model suggested in Section 3.5.2 was to assume that the soluble fraction is both the degradation-sensitive and the active fraction and is produced directly in proportion to the promoter firing rate. This means that the time evolution of the soluble and insoluble fractions can be treated independently. By this model, the production rate of *soluble* Tum could be estimated using the proportion of Tum that is soluble in the steady-state measurement of Tum. To settle on a number, an estimate of the ratio

of soluble to insoluble Tum at a similar production rate to *pR* was made using the data presented previously in Figure 3.21. By extrapolating linearly, it was possible to calculate the required level of IPTG such that the total mass of Tum in the CI^- degradation assay strain would match the 17 ng observed for the steady-state *pR* measurement. At that level of IPTG ($\sim$12 μM), the proportion of soluble Tum was about 33% of the total amount of Tum. By this estimate, then, the steady-state level of soluble Tum produced from unrepressed *pR* in the MFL strains becomes 8.3×10^3 molecules per cell. Then, using the short half-life of just the degradation-sensitive fraction, the production rate of the soluble fraction can be calculated as 2.1×10^3 molecules per minute, as listed in Table 3.4. Applying the converse analysis for the insoluble fraction would result in an effective production rate for that fraction of 403.6 molecules per minute.

For reference, a rate of production of Tum is also derived for the low-copy variant, pMTS*-*pR*-*tum*$^+$, based on the measurement for the medium-copy version. Plasmids derived from pSC101 have been variously reported with 6 [Xia et al., 1991], or 10–12 copies per chromosome unit [Lutz and Bujard, 1997]. For a doubling time of around 30 minutes, locus copy numbers per cell of between 1.5 for genes near the chromosomal termini, and 4.7 for genes near the origin are anticipated [Bremer and Dennis, 1996; Bipatnath et al., 1998]. With this consideration in mind, 20 copies per cell is taken as a representative figure for the copy number of pMTS-*pR*-*tum*$^+$, which then also represents the number of *pR*-*tum* modules per cell. For the pMTS*-*pR*-*tum*$^+$ plasmid with the SC101* origin, the literature reports copy numbers of around 25–40% of that of plasmids derived from pSC101 [Manen et al., 1994; Lutz and Bujard, 1997]; 30% is taken as a representative value giving 6 copies per cell for the SC101* origin. Using that estimate the unrepressed rate of production of soluble Tum from the pMCS*-*pR*-*tum*$^+$ plasmid can be estimated as 630 molecules per minute.

To judge the feasibility of the estimated production rates, the values given here in Table 3.4 are compared with the estimated firing rate of *pR* of 0.055 transcripts.s^{-1} Dodd et al. [2007b]. For the medium-copy plasmid there are around 20 copies of *pR* per cell, implying a combined *pR* firing rate of 66 transcripts.min^{-1}. Without any measurement of the half-life for these *pR* transcripts, the mean mRNA half-life of 5.2 minutes for *E. coli* [Bernstein et al., 2002] is assumed here so that there are an estimated 8.8 transcripts at steady state. Using the predicted steady-state concentration of soluble Tum, this means that approximately 900 proteins are present per transcript. This is greater than the median level of proteins per transcript observed by Lu et al. [2007] for *E. coli*, but still fits well within the log-normal distribution of such levels.

In the previous chapter (Chapter 2), a crude estimate of 7400 nM was assumed for steady-state production of Tum from from unrepressed *pR*; using the conversion factor derived in Section 3.4.4, that becomes $\sim$4100 monomers per cell. That estimate was derived assuming a chromosomal copy number for the *pR*-*tum* cassette and also a degra-

dation rate similar to that of CI. The medium-copy pMTS-*pR-tum*$^+$ plasmid has a copy number about 10–12 times that of the chromosome, but since the degradation rate of Tum is also 10 times faster than that of CI, the crude estimate can be compared 'as is' with the medium-copy measurement made here. The measurement of total Tum at steady-state (2.5×10^4 monomers per cell) is within ballpark range of the crude estimate. However, a fairer comparison may indeed be with the estimated 8300 monomers of soluble Tum, since that is the fraction thought to be degraded at the faster rate. That value compares much more favourably with the initial crude estimate of the production rate; given the moderate match between the modelling and experimental curves this lends some additional credence to the assumption of an active soluble fraction.

3.5.4 Production rate from P_{lac}

The P_{lac} promoter in the MFL serves to provide an IPTG inducible source of CI. To characterise steady-state concentrations of CI produced at each inducer concentration, the steady-state concentration for a single fixed concentration of IPTG is measured, and then this value is extrapolated to alternative concentrations of IPTG using the Hill curve fits of the induction reporter strains described in Section 3.3.

Since two alternative RBSs were used to access different ranges of CI concentration in the MFL strains, that is, the wild-type RBS (wRBS) in the pZC320-WR-*c*I plasmid, or the stronger pET plasmid-derived enhanced RBS (eRBS) in the pZC320-ER-*c*I plasmid, the steady-state level of CI for both of these plasmids needed to be measured. In both cases, the maximum level of induction used in the hysteresis assays (300 μM IPTG) was chosen as the reference concentration. Measuring the highest relevant *in vivo* CI concentration provided scope for optimising Western blot band intensities by diluting extracts into the most sensitive parts of the calibration curves. Without a clear understanding of the impact of the Tum−CI interaction on steady-state levels of CI, measurements of CI production were only made in the Tum$^-$ control strains.

Western blots of CI in cell extracts of the WR-MC and ER-MC Tum$^-$ MFL control strains were prepared alongside calibration extracts and are shown in Figure 3.25 along with example calibration curves. As described for *pR* production in Section 3.5.3, estimates of CI mass were made for each band using the calibration curves. Hill fits of the calibration curves generally fit quite well, but in two of the blots (blots A and B for the wRBS strain) there appeared to be an outlier in the calibration samples. In those cases, robust regression was used to objectively account for the possibility of outliers, and this resulted in better calibration as judged by improvements in internal consistency between each of the serial dilutions. The collection of mass estimates is shown plotted in Figure 3.26. Again, the greatest source of variation appeared to occur between cell extracts, so the confidence limits and values reported in Table 3.4 were derived from the mean estimates obtained for each extract.

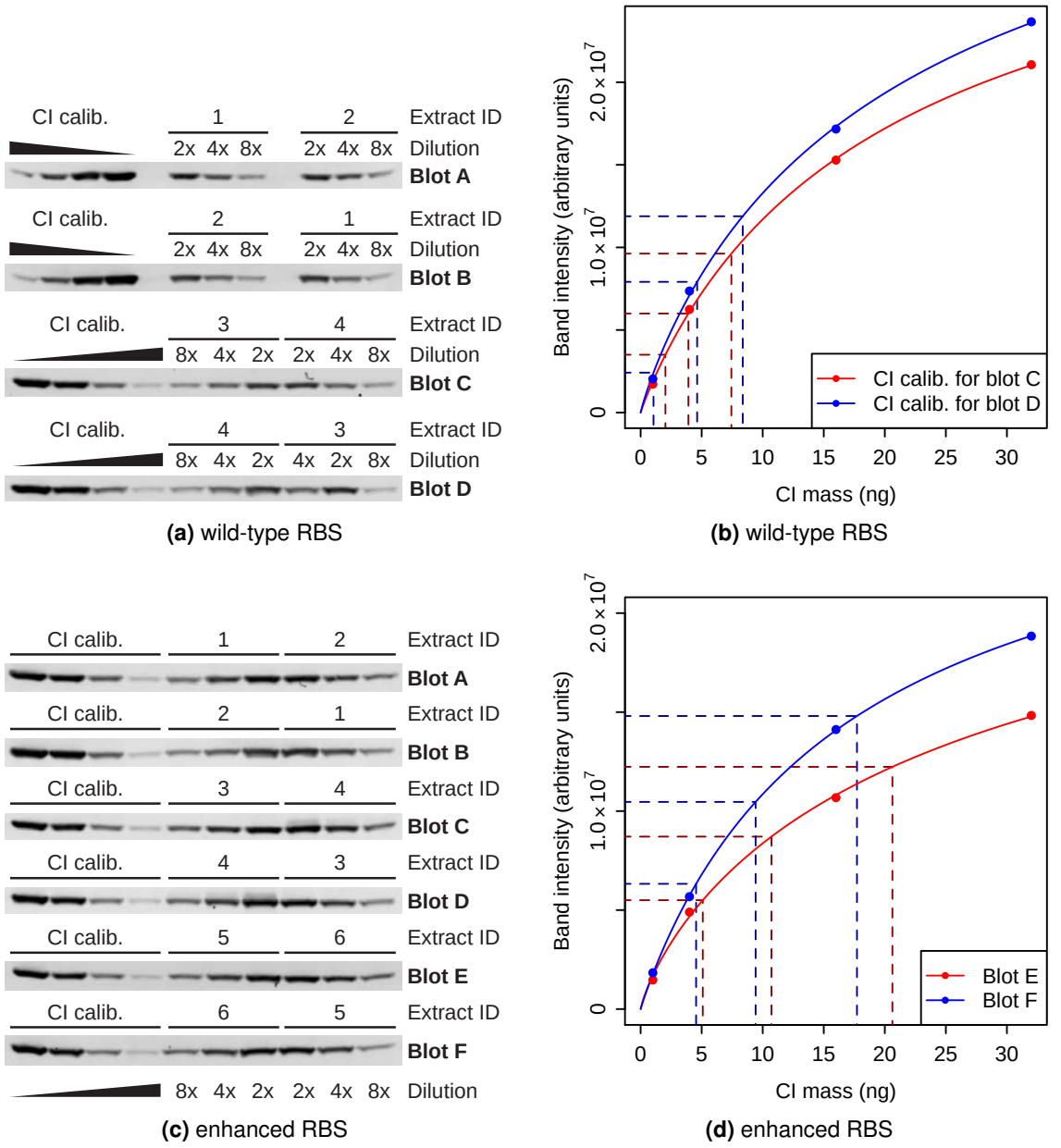


Figure 3.25: Quantitating steady-state CI production from P_{lac} by Western blot for induction at 300 µM IPTG. Cell extracts of log-phase cultures, diluted as indicated in equivalent CI^- extracts, and known amounts of purified CI were separated by SDS-PAGE, and transferred by Western for probing with α-CI antibodies (shown in (a) and (c)). 12 µL of each sample was loaded per lane. The blots were quantitated and steady-state CI mass in the cell extracts determined from a four parameter log-logistic fit (with basal parameter fixed to zero) of the calibration samples (examples shown in (b) and (d)). **(a)** & **(b)** Quantitating CI in cultures of E4300DR pZC320-WR-*cI* pMTS-*pR-tum*$^-$. **(c)** & **(d)** Quantitating CI in cultures of E4300DR pZC320-ER-*cI* pMTS-*pR-tum*$^-$. In (a) and (c), wedges indicate increasing dilution factors of the calibration samples, which were prepared from a master stock of 32 ng/12 µL of purified CI (made up in CI^- cell extracts), which was serially diluted to 1, 2, 8 or 32 times. In the example calibration curve fits, the dotted lines indicate the CI-specific band intensities of the dilution series of extract 3 (b) or extract 6 (d), and their corresponding mass estimates as determined by the fits.

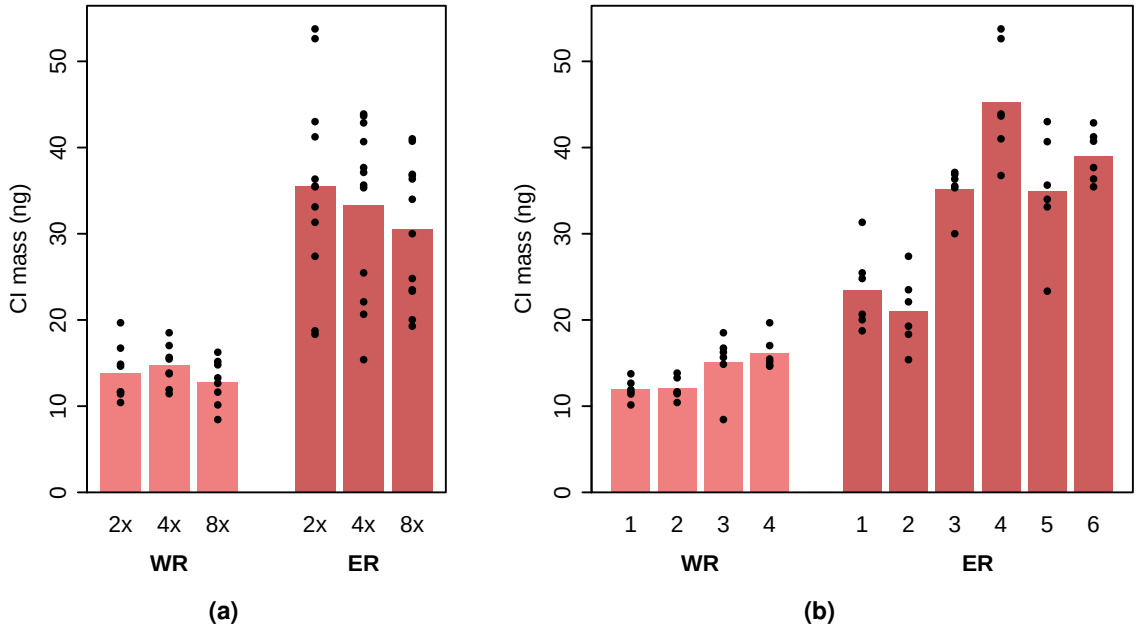


Figure 3.26: Estimates of the mass of CI in the wild-type RBS (**WR**) and enhanced RBS (**ER**) extracts using the calibration curves. The points show the individual estimates from each band of the blots. **(a)** Whilst the estimates are highly variable, the means of these estimates at each dilution indicate little trend in the data, giving greater confidence in the form of the calibration curve. **(b)** The mean estimates calculated for each extract. These values seem to cluster in pairs according to the groupings on the blots (see Figures 3.25(a) and 3.25(c)).

Cell concentrations at OD_{600} 0.6 were also measured for each of the strains, as was done for determining the Tum production rate (Section 3.5.3), and these are also listed in Table 3.4. Using these measurements and the known molecular weight of CI (21160 Da), the steady-state level of CI per cell for both the wRBS and eRBS variants could be derived. This revealed that with the eRBS, production of CI must be about 2.7 times stronger than for the wRBS. Since in Section 3.5.2 CI was not seen to be actively degraded, it is assumed to be lost only with cell division and growth. As determined in Section 3.4.2, this occurs with a doubling time (half-life) of 28.2 minutes for the *tum*$^-$ strains. Then in this case, by directly applying the steady-state formula for the production rate ($p = A\log(2)/h$), the production rates as listed in Table 3.4 can also be calculated.

For pZC320-WR-*c*I, steady-state levels of CI had also been previously measured by Western blot for induction levels ranging between 10 and 100 μM [Dodd and Egan, 2002]. At an induction level of 100 μM IPTG, 2390 monomers per cell were measured; the value measured here for production from pZC320-WR-*c*I, induced using only three times as much inducer (300 μM IPTG), is over five times higher than that. Even more alarmingly, production from P_{lac} has an ultrasensitive dependence on IPTG concentration and as seen in Section 3.3 has already started to saturate by 300 μM IPTG. Nonetheless, recall that the measurements made by Dodd and Egan [2002] were used earlier in Figure 3.11 to scale the P_{lac}-*lacZ* induction reporter curves to units of CI monomers per cell. By similar extrapola-

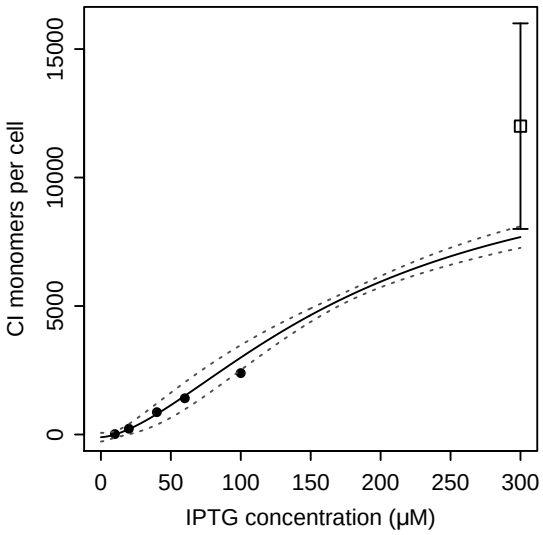


Figure 3.27: Comparing steady-state estimates of CI concentration for the MFL strains with those previously obtained for the same induction system. Shown plotted versus the concentration of IPTG inducer are steady-state measurements of CI concentration in cells with the pZC320-WR-*c*I and pUHA-1 plasmid induction system [Dodd and Egan, 2002] (filled black circles); plotted alongside is the estimate obtained here for the WR-MC *tum*$^-$ MFL strain induced using 300 μM IPTG with 95% confidence limits (open square). Overlaid on this data is the Hill curve fit obtained for the low O/N Tum$^-$ induction reporter assay from Figure 3.8(b). Dotted lines indicate 95% confidence limits for the fitted Hill curve.

tion using the induction reporter Hill curve fits, a comparison can be drawn between the estimates of CI levels obtained by Dodd and Egan [2002] and those obtained here. Using the correlation between the Western and LacZ assay data which ignored the maximum data point (see Figure 3.11(a)) and taking 95% confidence limits into account, it can be seen in Figure 3.27 that the apparently large discrepancy between the two assays may not be significant. Furthermore, the strain used by Dodd and Egan [2002] did not contain the additional tetracycline resistant MFL plasmid that the MFL strains have. So, as discussed in Section 3.4.2, the rate of growth would be likely to be slower in the MFL strains. Then since loss of CI is dilution limited, the relative increase in the observed steady-state level of CI would come as less of a surprise.

3.6 Chapter summary

By cloning the Tum−CI MFL regulatory network in *E. coli* using a modular design, an iterative optimisation of the synthetic circuit was possible. Then using a process of elimination, a set of four candidate strains was obtained and assays of those strains revealed that the system can exhibit the predicted bistability. However, the degree of hysteresis seemed small in comparison with the modelling predictions, and in some cases was an incomplete loop. The divergence prompted a closer look at the behaviour of some of the components involved in the circuit, including the form of dependence of the P_{lac} promoter on the concentration of its inducer, the growth rates of the strains and the production and degradation rates of Tum and CI. By making use of the greater understanding such analysis brings, the behaviour of the MFL can be reassessed and optimised. In the coming chapters, this data will become invaluable in better matching the experimental observations of the Tum−CI MFL to an evolving mathematical model that describes this bistable circuit.

4

The MFL displays only weak bistability when measured over a whole population

Using a modular and flexible circuit design, four synthetic Tum−CI MFL gene circuits that were promising candidates of bistable behaviour were cloned in *E. coli*. A hysteresis assay of these circuits was chosen as the primary test for bistability, since it would enable both analysis of stability and efficient optimisation of a critical parameter to expedite the search for a parameter regime supporting bistability. However, unlike the hysteresis curves predicted from the model developed in Chapter 2, the experimental hysteresis curves did not show any clear boundaries between monostable and bistable regions, thus limiting the power of the assay to definitively assess bistability. In this chapter, by making use of the additional conceptual and quantitative constraints provided by the characterisation of system components in Chapter 3, the whole-population assays of hysteresis are analysed and extended so that a reliable assessment of bistability can be made by direct comparison between the experiment and model.

Since the hysteresis assay is used here as the key indicator of bistability, the first part of the chapter is devoted to a discussion of the experimental factors that affect the accuracy and resolving power of the assay itself. Then, whilst the control curves are expected to display some hysteretic behaviour (refer back to Sections 2.4.1 and 2.4.2), the separation of both the controls and MFL curves at high CI production rates prompts a deeper investigation into the persistence of hysteresis: a bistable system should exhibit hysteresis that persists over time. The chapter concludes by exploring the extent to which the deterministic model can describe the data observed.

4.1 Experimental limitations of the hysteretic LacZ assay

Although the preliminary experimental results revealed significant separation between the hysteresis curves of the MFL strains, there were two factors, in particular, that obscured clear judgement of bistability in the MFL: (1) a larger than normal variance in the measured LacZ units, and (2) a poorly defined shape of the hysteresis curves. This section explores the extent to which these obscuring factors can be explained as technical artefacts of the assay. The first of the points will be addressed by analysing the data obtained from the

hysteresis assays to show that LacZ units have a nontrivial dependence on the final OD_{600}. For the hysteresis assays, there is an increased variability in cell densities measured at the time of assay that makes this effect more significant. The dependence of LacZ units on OD_{600} can be corrected for by approximating the trend and these corrections explain much of the increased assay variability.

The second point is addressed by drawing upon the efforts made in Chapter 3 to characterise the P_{lac} induction circuit and the corresponding production and degradation rates of CI. By expressing the hysteretic variable in terms of the calibrated production rate of CI, the added nonlinearity due to the induction circuit is removed. This makes for a much fairer comparison with the models introduced in Chapter 2, and results in a clearer picture of the region of bistability.

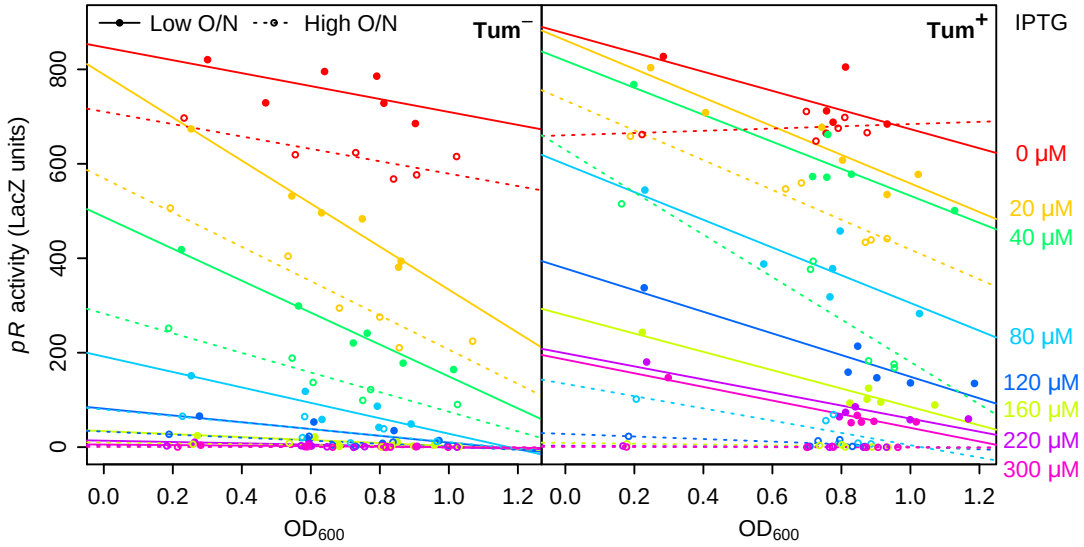
4.1.1 Variations in optical density bias LacZ assay measurements

To be confident in the separation between two curves, the assay used to measure such curves needs to have sufficient discrimination power. In the hysteresis assays of the MFL strains, excessive experimental noise could obscure an underlying hysteresis cycle if the variance in measurement of *pR* activity becomes too large. Larger than expected variation of LacZ units was found for the MFL strains measured at the normal equilibration time and, as the limits of this assay are pushed to optimise the time for equilibration, the resolving power of the LacZ assay could (and does) become an issue. Identifying and reducing sources of experimental noise will increase the value and usability of the assay for judging bistability.

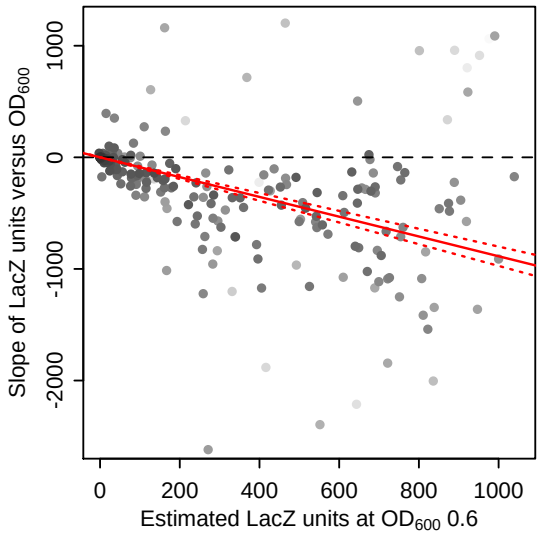
As a population assay, the LacZ assay relies on an accurate measurement of cell density to ensure consistency of results. For practicality, the OD_{600} of the assay cultures is used as the measure of cell density and, in calculating LacZ units, the β-gal enzyme activity measured for the culture is divided by the OD_{600}. For moderate variations in final culture OD_{600}s this has proven to be a suitable correction, but in the hysteresis assay, variation in final OD_{600} was found to be larger than typical. This is presumed to arise as a result of accumulated pipetting errors due to the additional dilution step required to get the normal (as opposed to the short) equilibration time. This increased variability in final OD_{600} revealed an unanticipated trend for samples with a low OD_{600} to produce higher LacZ units on average than samples with a high OD_{600}. This trend may well be due in part to the nonlinear relationship of cell mass density with the OD_{600} Bipatnath et al. [1998].

To characterise and confirm the result, the LacZ units measured within each experimental factor of the MFL hysteresis assays were plotted against the OD_{600}. The dependence of LacZ units on OD_{600} was approximately linear, but the slope of each model seemed to have a nontrivial dependence on the magnitude of the activities. The trend is described in more detail in Box 4.1 along with a method for correcting for the deviation. By correcting for this effect, the LacZ units are made more consistent, both within each

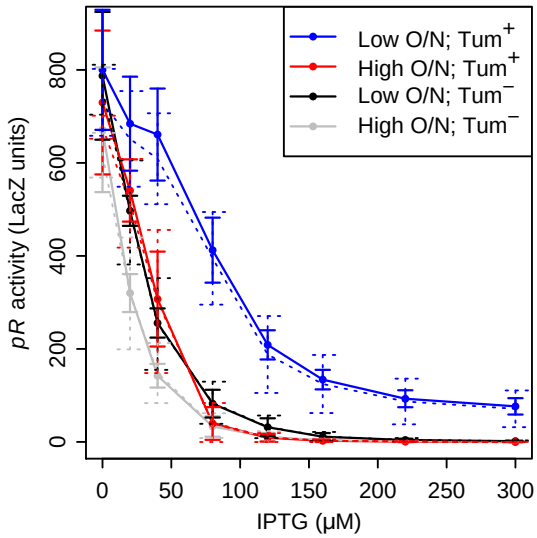
Box 4.1: Correcting for the dependence of LacZ units on OD_{600} reduces variability and increases separation between the hysteresis curves. By examining the relationship between LacZ units and OD_{600} for each factor of the hysteresis assays, higher OD_{600}s are found to produce lower LacZ units, as exemplified in Panel A. The dependence on OD_{600} is a nontrivial function of the intrinsic reporter activity, but generally shows a stronger effect at higher activities (Panel B). A linear approximation of this trend allows estimation of the typical dependence on OD_{600} for each LacZ measurement, enabling transformation of the raw data to values predicted for an OD_{600} of 0.6, as shown applied in Panel C.



A LacZ units measured from hysteresis assays of the ER-MC MFL strain are shown plotted against their respective OD_{600}s. Data from the Tum^- control strains and Tum^+ MFL strains are shown separately. Data from the low or high O/Ns are distinguished as indicated, and the IPTG concentrations are coded by colour as per the labels. Linear regression was performed individually for each experimental condition in the assay to produce the lines of best fit.



B Regression lines like those in Panel A were derived for all hysteresis assays in this chapter and their slopes are plotted here versus their values at OD_{600} 0.6. The standard errors of these fitted slopes were used as weights for weighted linear regression of the data (solid red line; dotted lines indicate 95% confidence intervals). The points are shaded in grey according to the weights with lighter shades having lower weights/larger errors. The dashed black line marks a slope of zero.



C The raw hysteresis data obtained for the ER-MC strains (dotted lines and error bars) is shown compared with the same data after correction for the dependence of LacZ units on OD_{600} (solid lines and error bars). The correction slope for each LacZ measurement was estimated from the regression line in Panel B and that slope used with the respective final OD_{600} reading to estimate the LacZ value at OD_{600} 0.6. Error bars indicate 95% confidence limits.

experimental condition, but also across different experimental conditions, as judged by stabilisation of the mean values with respect to general data trends[1]. The correction also increases the separation between the hysteresis curves in two ways: firstly by decreasing sample variance, hence increasing significance, secondly, and unexpectedly, by shifting the mean values further apart. The trend of LacZ units with the OD_{600} will be corrected for in a number of assays in this thesis; where the correction is applied, the resulting data will be referred as having 'Corrected LacZ units'.

4.1.2 Normalising P_{lac} induction levels to production rates improves but does not complete the picture of hysteresis

The concentration of IPTG in the culture controls the rate of production of CI in the MFL strains, and the hysteresis assay described in Section 3.2 is based on controlling this production rate by adjusting the IPTG concentrations over time. However, whilst IPTG concentration is a quantitative measure, it does not translate directly into a parameter of the deterministic MFL model described in Chapter 2. As seen using P_{lac}-*lacZ* assays in Section 3.3, intracellular concentrations of the P_{lac} promoter's gene product have an ultrasensitive dependence on IPTG, attributed to cooperative binding of the LacI repressor. This nonlinear relationship is likely to directly affect the interpreted locations of the bifurcation boundaries by introducing complex changes in curve shape. By utilising the induction curves and other parameter determination experiments of Chapter 3, these shape-changing effects can be corrected for and the data expressed in terms of a variable that relates directly to the modelling: the CI production rate.

The conversion from each experimental condition (i.e., idealised final IPTG) to CI production rate is calculated in a number of steps, making use of many of the measurements reported in Chapter 3:

1. The final concentration of IPTG is corrected using Equation (3.1) (Section 3.3) to account for the IPTG concentrations contributed by the subcultured O/Ns.
2. The Hill fits of the P_{lac}-*lacZ* induction curves are used to normalise these concentrations to LacZ activity (a linear measure of the rate of production from P_{lac}), taking the Hill parameters from Table 3.1 for the 'Mixed' data sets (i.e., those for which low and high O/N data sets were aggregated).
3. These relative production rates are scaled to absolute CI production rates in dimers per minute using the experimentally determined CI production rates listed in Table 3.4 for growth in media with 300 μM IPTG[2].

[1] This latter improvement was less obvious for the example given in Box 4.1, Panel C than it was for the long equilibration time assays to be introduced later in Section 4.2.2.

[2] Note that in this case, the final IPTG concentration is exact since log growth cultures were subcultured from high O/Ns in the steady-state measurements of CI concentration. As such, the induction curves are scaled to give the listed production rates at exactly 300 μM IPTG.

Note that the above process results in a different normalisation depending on the strain, since the Hill parameters differ between *tum*$^+$ and *tum*$^-$ strains and the CI production rates differ between wRBS and eRBS strains.

Applying this induction normalisation to each of the normal equilibration time hysteresis curves (previously shown in Figure 3.6), results in curves whose shapes now better emphasise the putative regions of bistability (see Figure 4.1). This is made manifest in a couple of ways. Firstly, the separation between the MFL curves and the control curves is greater, making the putative monostable region at low CI production rates more evident. Secondly, the regions with the greatest separation between high and low O/N MFL curves now occupy a greater fraction of the hysteretic variable domain. This is especially obvious for the strains with medium-copy *pR-tum* plasmids. In the strains with low-copy *pR-tum* plasmids this effect is far less pronounced, highlighting instead the collapse of those curves towards an average value. This makes it hard to propose a region of bistability for the low-copy MFL circuits, casting in doubt whether they are in fact bistable. Relative to the deterministic model curves, the medium-copy MFL assays also show some collapsing towards the mean, but are far more likely to contain a region of bistability. The remaining discussion will focus on these curves.

For the medium-copy MFL, a significant separation was observed between the high and low O/N hysteresis curves at high production rates, and normalising to a CI production rate does not change this. However, the normalised curves do appear to compare more favourably with the deterministic MFL model. In the model, the low O/N curve asymptotes towards the high O/N curve at high CI production rates to form the high monostable state (see Figure 2.12). The raw IPTG response curves were at odds with this, seemingly asymptoting towards the significantly different value occuring at maximum induction. After normalisation to the CI production rate, these curves instead appear more as if truncated within the bistable region, leaving scope for a return to the high O/N curve if higher CI production rates were reached. Nonetheless, even if such a trend were to be found, it remains an anomaly that the high CI monostable state seems to be missing from these hysteresis curves. Without that monostable state, the model is incapable of explaining how the 'off' curve (i.e., high O/N curve) can be accessed. Instead, an explanation in terms of the physiological differences between stationary phase (O/N) and log phase cultures must be resorted to. An alternative explanation may be that at maximum induction, the medium-copy MFL is actually in a monostable region, but slow equilibration at that end has prevented the curves from coinciding; such an explanation is more likely for the ER-MC strain.

The Tum$^-$ control curves are useful in providing a sense of how far from equilibrium each system is, and it was noted for the IPTG response curves that these controls showed hysteretic behaviour by not yet reaching equilibrium. Normalisation reveals that these Tum$^-$ curves overlap better than previously realised, particularly for the wRBS strains

where it appears as if equilibrium has been reached. The slower equilibration of the eRBS strains is not unexpected since the larger concentrations of CI result in longer times for reaching steady state. Perhaps more curiously, the control strains with low-copy MFL plasmids appeared further from equilibrium than those with the medium-copy MFL plasmids. The low-copy strains generally showed longer doubling times (see Table 3.2), so perhaps in these strains equilibration takes longer for the dilution-limited CI and β-gal proteins. Nonetheless, the significant separation of the ER-MC control curves suggests that in the MFL strains, more equilibration time may be necessary to see closure to a monostable state at high CI.

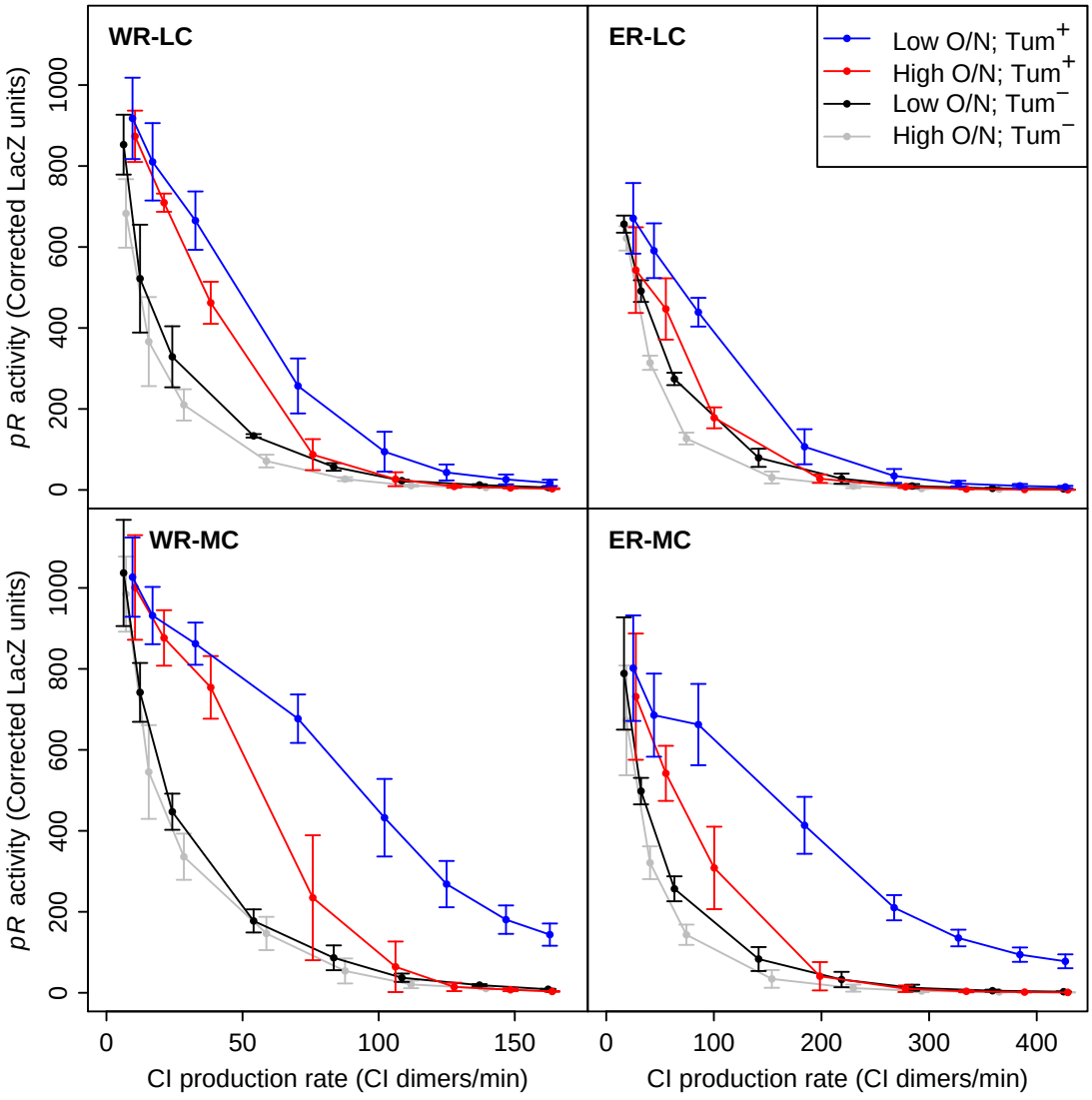


Figure 4.1: Expressing the normal equilibration time hysteresis curves in terms of CI production rate reveals a wider putative region of bistability. The raw data from Figure 3.6 is reproduced here with corrections to the LacZ units for trends with OD_{600} as described in Box 4.1 and calibration of IPTG concentrations to CI production rates as described in the text. Normal equilibration time hysteresis assays were performed for four different MFL strains as labelled; translation of CI was either from its wild-type RBS (WR) or an enhanced RBS (ER), and the *pR-tum* module was either located on a low-copy plasmid (LC) or medium-copy plasmid (MC). Assays of the respective Tum^- control strains are also shown. In the hysteresis assay, assay cultures are innoculated either from overnight (O/N) cultures grown in the presence of 0 µM IPTG (low O/N curves), or 300 µM IPTG (High O/N curves). Error bars show 95% confidence limits in the mean; $n = 6$ for all data points except for the high overnight Tum^+ curve of the ER-LC strain, which has $n = 4$ for all data points.

4.2 Extending the hysteresis assay equilibration time

After setting the MFL in either of its two initial states, a certain amount of equilibration time must be allowed for the system to settle towards its steady state. The importance of such equilibration time is clear from the predicted and observed behaviour of the monostable controls: hysteresis assays of these Tum$^-$ controls do show history-dependent behaviour. However, when sufficient time is allowed, a monostable system should get close enough to steady state that the two hysteresis curves become indistinguishable. So to show bistability for the MFL strains, enough time should be allowed to be confident that separation of the hysteresis curves is persistent.

Ultimately the choice of how much equilibration time to allow is somewhat arbitrary, especially considering that the time taken to reach steady state near the bifurcation boundaries is predicted to extend indefinitely (see Section 2.4.2). However, in a more practical sense, the equilibration time should at least be long enough to distinguish between hysteretic behaviour within the bistable region and the anticipated non-hysteretic behaviour in the monostable regions located to either side. Such a contrast was not obvious for the normal equilibration time curves shown in Figure 4.1, where it was unclear if the strains with the low-copy *pR-tum* plasmid showed a bistable region, or if the strains with the medium-copy *pR-tum* plasmid showed a high CI monostable state. By further increasing the time for equilibration such uncertainties can be tested.

Since the Tum$^-$ control strains provide some measure of the time it takes for an equivalent monostable system to resolve to steady state, this section starts by exploring different equilibration times for hysteresis assays of the control strains. The long equilibration time from those trials is chosen as a good balance between overlapping monostable control curves and a practical time frame. When applied to the MFL strains, a different picture of hysteresis and its persistence emerges. The low-copy MFL strains no longer show a region of bistability and this region all but disappears for the medium-copy strains. Whilst this result might imply that none of the circuits show bistable behaviour, some unresolved discrepancies provoke an explanation favouring some limited level of bistability.

4.2.1 Extending the time for equilibration brings the control strains to steady state

The Tum$^-$ control strains include no means for positive feedback and are thus monostable systems representative of each respective MFL variant. As monostable circuits, these control strains are not expected to show persistent hysteresis. Instead their hysteresis curves should tend towards a single steady state. However, in the experimental hysteresis curves shown in Figure 4.1, the controls still show significant separation in some cases. In particular, the control strains using the eRBS for CI production showed more separation than their wRBS counterparts. Since the MFL strains with medium-copy *pR-tum* plasmids

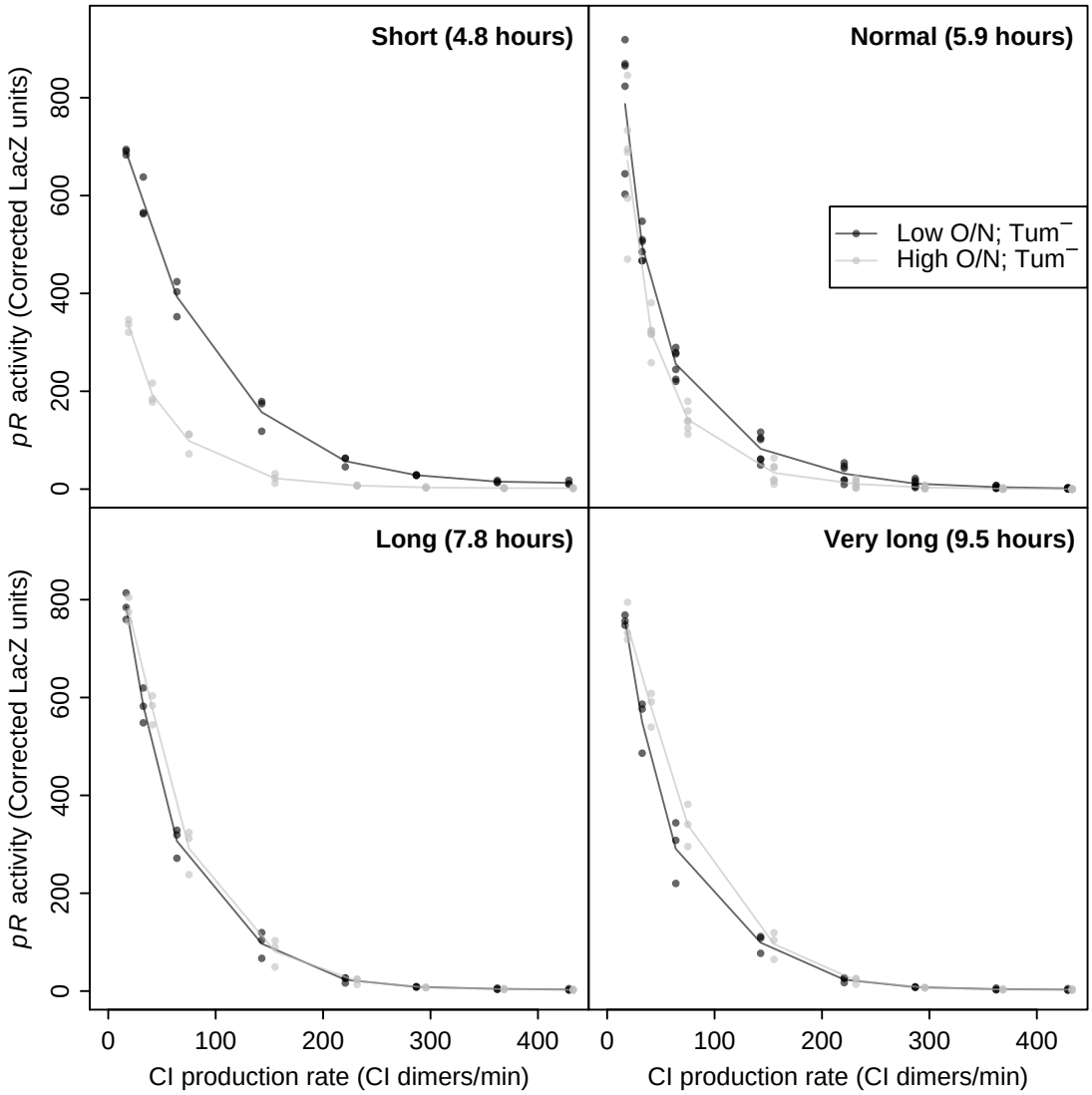


Figure 4.2: Extending the time for equilibration allows the control curves to reach equilibrium. Hysteresis assays were performed for cultures of E4300DR pZC320-ER-*c*I pMTS-*pR-tum*$^{-}$ with either short (no additional dilution), normal (with additional 1/10 dilution), long (no additional dilution but subcultured twice), or very long (with additional 1/10 dilution and subcultured twice) equilibration times. The normal equilibration time data is taken from Figure 4.1. The time of measurement is listed at the top right of each panel. Each point represents an individual measurement; lines follow the averages. The LacZ units have been corrected as per Box 4.1 and the final IPTG concentrations normalised to production rates of CI as per Section 4.1.2.

were more promising candidates for bistability, the ER-MC Tum^{-} control strain was thus chosen as a model system to test how much equilibration time would be necessary to reach an equilibrium result in a monostable circuit similar to the MFL.

As previously introduced in Section 3.2, the equilibration time for the standard LacZ assay can be extended by increasing the time taken for the cultures to reach the target OD_{600} 0.6 for the LacZ assay. The normal equilibration time assay adds an extra 1/10 dilution of the O/N cultures before the final subculturing step to increase the number of

cell doublings required to reach log phase. To further extend the equilibration time, larger dilutions were trialled (1/20 and 1/40), but this approach was rejected on the basis of much noisier assay results, presumably by amplified pipetting errors and culture inhomogeneity. Instead, an alternative approach was trialled on the model ER-MC Tum$^-$ control strain, whereby subculturing of the hysteresis assay cultures was repeated midway through the growth to log phase. This additional subculture step involved a 1/50 dilution of all log growth cultures into a duplicate log growth plate once an average OD_{600} of approximately 0.06 was reached. This approach was trialled alternately to log growth cultures started with or without the extra 1/10 dilution of the O/N cultures, with these new protocols being named 'very long' or 'long' equilibration time assays respectively.

Short, normal, long and very long equilibration time hysteresis assays were carried out for the ER-MC Tum$^-$ control strain and the results are shown in Figure 4.2. As anticipated, the time taken for cultures to reach the target OD_{600} increased with increasing dilution factors. The additional subculturing step of the long equilibration time assay extended the time of assay by almost 2 hours[3] compared with the normal equilibration time assay; in both normal and very long equilibration time assays, the additional dilution of the overnight cultures extended the time of assay by over an hour.

With longer equilibration times, the control curves get closer to their expected steady state values. The short equilibration time assay leaves the control curves clearly separated, and whilst this separation is greatly reduced for the normal equilibration time curves, significant separation between the two curves still remains. The deterministic model predicted that the greatest separation would be observed for the lowest CI production rates, but curiously separation in these curves appears greatest at intermediate CI production rates. Possible explanations may be the increased experimental error at high *pR* activities, or the closer proximity of CI production rates at this point[4]. In any case, by the measurement time of the long equilibration time assay, the two curves are overlapping, and by the measurement time of the very long assay the high O/N curve is *above* the low O/N curve. This seeming overshoot of equilibration could just be experimental error, but may also suggest that corrections in the normalisation to CI production rates could more accurately portray that assay[5]. The very long equilibration time assay is still taken to be the most equilibrated.

The time to equilibration of the control strains is limited primarily by the degradation rates of CI and β-gal, whereas equilibration of the MFL strains is additionally limited by the degradation rate of Tum (refer back to Section 2.4.2 for details). This means that the minimum equilibration time needed for the controls is only a lower bound on the time

[3]With only one replicate for each assay type, the indicated timings are only rough estimates since there is some variation in the average final OD_{600} of each experimental set.

[4]Another explanation may be some nonlinear dependence of LacZ units on activity for high LacZ unitss.

[5]In the assays with an additional subculturing step, the absolute concentration of IPTG changes slightly during the course of the assay, since subculturing the log growth cultures into fresh assay media further dilutes the relative offsets produced by the IPTG concentrations of the O/N cultures.

required to demonstrate bistability of the MFL. Nonetheless, since the long equilibration time assay produced control curves that are indistinguishable, that assay was chosen as the most practical and appropriate next candidate.

4.2.2 Complete hysteresis is observed with a long equilibration time

The MFL strains exhibited a degree of hysteresis when assayed using a normal equilibration time, but the existence of a bistable region was left in question by a number of disequilibrium artefacts. By assaying these strains at a longer equilibration time, the persistence of hysteresis can be assessed which should further enable a better rationalisation of the steady states and regions of bistability and monostability. As discussed in the previous section, the long equilibration time was sufficient to show monostability for the ER-MC Tum$^-$ control curves, so this time point should also be a good reference point for a qualitative assessment of separation between the MFL hysteresis curves. Based on the long equilibration assay trial, hysteresis assays of the MFL strains were performed which also incorporated an extra subculturing step partway through growth to log phase. For improved consistency, this second subculture step was always performed 4 hours after subculturing from the overnight cultures; by 4 hours cultures had an average OD_{600} 0.2. The results of those assays for the four candidate MFL strains are shown in Figure 4.3.

As previously observed in Section 4.2.1, the long equilibration time is sufficient to bring all of the Tum$^-$ control curves in Figure 4.3 to a single monostable steady state. More importantly, the longer equilibration reveals that at the highest CI production rates for the ER-MC MFL strain, the low O/N curve does close to the high O/N curve. Thus assays of the ER-MC MFL include both the low CI and high CI monostable regions, so that complete traversal of the hysteresis loop can be observed in that strain. In contrast, and as predicted for even longer equilibration times found from preliminary hysteresis assays of a similar MFL strain [Pocock, 2007], the WR-MC MFL strain continues to show significant separation at its highest CI production rate. The most likely explanation is that the WR-MC MFL strain is indeed still bistable at the highest levels of CI that can be accessed in log phase growth. However, the change to stationary phase in the O/N cultures must enable still higher concentrations of CI that are then able to set the switch in the putative high CI monostable state. Nonetheless, all hysteresis loops do show signs of better equilibration and the only unclosed loop is that of the WR-MC strain. In spite of this, the more striking (and alarming) result from the long equilibration time assays is a greatly reduced separation between low and high O/N curves in *all* MFL strains.

Separation of the hysteresis curves is somewhat more favourable for the MFL strains with medium-copy *pR-tum* plasmids than for those with low-copy ones, but the clear separation previously seen for the normal equilibration time point is no longer apparent. A visual inspection of error in the WR-MC hysteresis curves indicates significant separation of the high and low O/N curves, but a similar inspection of the ER-MC curves indicates

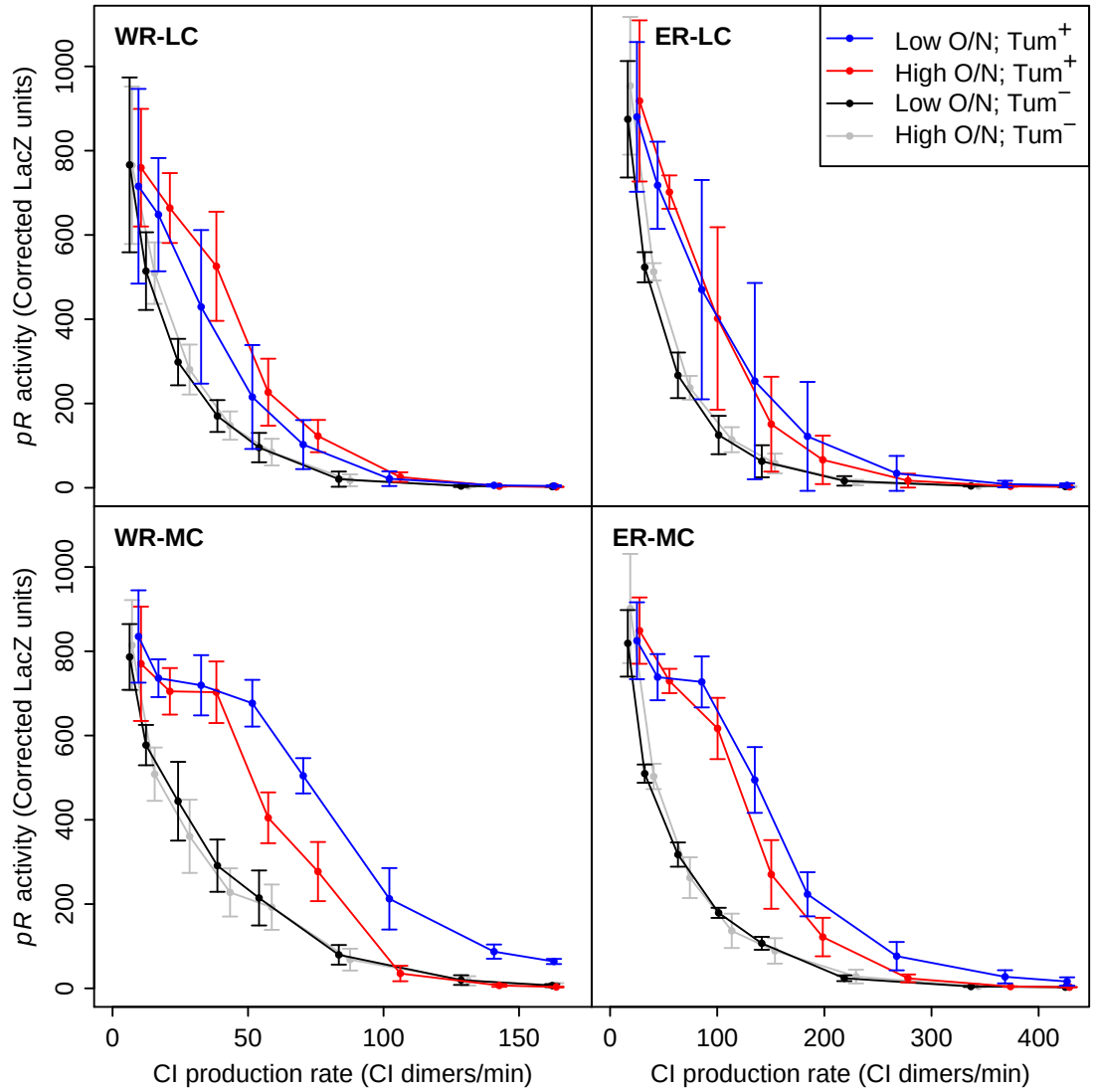


Figure 4.3: Complete hysteresis is observed when using the long equilibration time assay at the cost of loop collapse. Hysteresis assays were performed with a long equilibration time as described in Figure 4.2 for the candidate MFL strains described in Figure 4.1. Data normalisation is also as described in those figures. Error bars show 95% confidence limits in the mean; $n = 6$ for all data points.

separation is statistically nonsignificant. To consequently draw the conclusion that the WR-MC MFL has a region of bistability, where the ER-MC MFL does not, would, however, conflict with the assumptions of the deterministic model. In terms of the steady-state model, the only differences expected from a change in CI RBS strength are rescaled CI production rates. Such a change should only affect which portion of the steady-state curve is visible to the assay; the absolute CI production rates at which the system is in bistable or monostable regimes should not change. When out of steady state, the system will be head over time towards the nearest steady state. Hence, after assuming that the initial states for all strains are situated in opposing monostable regions (an assumption supported by the experiments), then the CI production rates at which the WR-MC MFL is bistable should

show similarly persistent bistable behaviour in the ER-MC MFL[6]. The preferred conclusion is thus that the ER-MC MFL is capable of bistability, but that experimental errors or unidentified sources of additional intrinsic noise mask the bistable region of this strain.

The strains with low-copy *pR-tum* plasmids no longer show any identifiable separation. In fact, the high and low O/N curves for the WR-LC MFL appear flipped with respect to their expected locations. This shows that cultures of the low-copy *pR-tum* MFLs cannot, as an entire population at least, stably maintain the initial state — there is no region of bistability apparent in these strains. The trajectory towards this effective monostable steady state in these MFL strains is still slower than for the Tum$^-$ controls, but the apparent hysteresis loop does not persist that much longer than the time taken for the controls to reach equilibrium. Nonetheless, an unusually large concomitant increase in culture-to-culture variation for the low-copy *pR-tum* MFL strains gives reason for further investigation before bistability is conclusively rejected in these cases. The increase in error does not seem to be experimental in origin since assays are performed as a complete set in 96-well plates, and the control curves do not show the same increase. However, if not experimental then the additional noise must be heritable over a number of generations since it must be apparent at a whole population level. One explanation may be an unstable inheritance or copy-number of the low-copy *pR-tum* plasmid. In any case, persistent hysteresis for the low-copy MFL strains does not occur over this time scale; in these strains, the rate of production of Tum from *pR* is presumed insufficiently strong to outcompete the levels of CI.

The results of the long equilibration time hysteresis assay reveal that persistence of hysteresis in the four MFL strains is much weaker than realised from the normal equilibration time hysteresis assays. The strains with low-copy *pR-tum* plasmids no longer show any hysteresis, whilst the strains with medium-copy *pR-tum* plasmids show limited hysteresis, though it seems that noise or error obscures this for the ER-MC MFL. The collapsed shapes of these latter hysteresis curves are, however, distinct from the simple deterministic model curves seen in Section 2.4.2 for assumed parameters. An important next step towards understanding the nature of bistability in the medium-copy strains will be to update the model and its parameters to better reflect the experimental curves.

[6] Clearly when judging CI production rate by the axis labels in Figure 4.3, there are significant discrepancies in curve shape and value between WR-MC and ER-MC strains. However, since the mismatch also pertains to the Tum$^-$ controls, the more likely deficit is the significant error in the experimental estimates of CI production rates. Corrections to the relative scaling of these production rates will be considered when modelling the curves in Section 4.3.

4.3 The deterministic model does not capture the behaviour of the MFL

The experimental curves seen in Figure 4.1 for the normal equilibration time and in Figure 4.3 for the long equilibration time do not bear an obvious relationship to the model curves characterised in Chapter 2. This seeming deviation from the model makes it hard to draw definitive conclusions about the existence and/or locations of the putative regions of bistability in the experimental MFLs. The model curves were generated using a set of parameters, many of which had to be estimated or were determined with considerable measurement error. These uncertainties leave scope for parameter variability and permit a degree of flexibility in the shape and form of the resulting model curves. In this section, the capacity of the current deterministic model to explain the available experimental data through such parameter variation is assessed. The section begins by developing methods for exploring parameter variability in the deterministic Tum−CI MFL model and assessing the model against the available data. These methods are then applied to find optimum parameters that best fit the deterministic model to the data obtained for the WR-MC MFL and control strains. By making use of that preliminary fit, the data obtained for both the WR-MC and ER-MC strains can then be reconciled and an optimum parameter set found that best fits the model to this combined data set. This extensive analysis reveals that parameter variation in and of itself is insufficient to accurately describe the behaviour of the Tum−CI MFL, and calls for a rethinking of the model itself.

4.3.1 Searching the parameter space of the deterministic MFL model

Obtaining reliable estimates for model parameters is a common challenge faced in the quantitative modelling of gene networks [Karlebach and Shamir, 2008]. In the case of the deterministic Tum−CI MFL model, a simple substitution of the parameters estimated from the literature in Chapter 2 with the more realistic experimental measurements of parameters that were reported in Chapter 3 does not improve the match between experiment and model. This does not come as a great surprise: large uncertainties of 30–50% were found for a number of parameters, whilst other parameters (those for the Tum−CI equilibrium) still needed to be derived from *in vitro* measurements, which could well differ *in vivo*. An attractive alternative is to infer the parameters by fitting the model to suitable data. Such model-driven parameter estimation has been used for studying many gene networks, from the regulation of lysis and lysogeny in phage λ [Shea and Ackers, 1985; Dodd et al., 2004] and phage 186 [Dodd et al., 2007b], through synthetic networks constructed in *E. coli* [Guido et al., 2006] and yeast [Ellis et al., 2009b], to receptor-mediated apoptosis in human cells [Eydgahi et al., 2013]. Fitting the deterministic Tum−CI MFL model to the LacZ assay hysteresis data will provide refined parameter estimates and, more importantly, a thorough test of model suitability. The parameter estimates of Chapter 3 remain

useful as 'sign posts' for setting initial guesses and reasonable parameter bounds.

With the additional goal here of also testing the suitability of the model, a reasonably thorough search of the parameter space is desirable to ensure that good fits of the data are not missed. However, even for the simple steady-state model of the Tum−CI MFL with only 7 parameters, it is a nontrivial exercise to search the resulting 7-dimensional parameter space for values that might produce fits of the experimental data. Separately analysing the effect of each parameter (as was done for the steady-state bifurcation boundaries in Section 2.3.2) will leave a majority of the parameter space out of the analysis, whereas checking all possible parameter combinations is impractical and even computationally unfeasible. To make the problem more tractable, the divergence of the model from the data is described by a 'cost function' — a function that scores how well the model fits the data for any given parameter set, typically a sum of squared residuals — and then well-established optimisation routines can be employed to intelligently navigate this function's landscape to find the global minimum. The set of parameters producing this global minimum would mark the best match of the model to the data; better matches should in theory only be possible by modifying the model.

Unfortunately, cost functions typically have very complicated landscapes with numerous local minima that make it hard to find the global minimum. To maximise the extent of the search without exceeding manageable computation times, a pseudo-random search of the parameter space is used to search the cost function landscape. The analysis here is facilitated by the R package FME [Soetaert and Petzoldt, 2010], which provides a framework for constructing the cost function and implementing the search algorithm. The package makes a number of parameter optimisation routines available, but the pseudo-random search algorithm of Price, as described in Soetaert and Herman [2009], is the favoured method for good sampling of the parameter space. The algorithm requires specification of upper and lower limits for each parameter, and starts by randomly generating a population of around 50 candidate sets of model parameters between these limits. It then takes a biased random walk in the 'vicinity' of this population, iteratively updating the population as parameters that reduce the residuals are found. The algorithm terminates either when variation in the cost function amongst the population is reduced below some threshold, or when it reaches 10000 iterations.

For the Tum−CI MFL hysteresis data, a *weighted* least-squares cost function was chosen, since larger LacZ units tend to be accompanied by increased standard deviations. In other words, the results with higher activities should have less influence on the fit, that is, they should be weighted less heavily. Often the weight of each residual is set to the inverse of the standard deviation. However, with only 6 data points per experimental factor in the Tum−CI assays, the sample standard deviations (SDs) are a biased choice. Instead, a log-log regression of SD versus sample mean was found to serve as a better in-

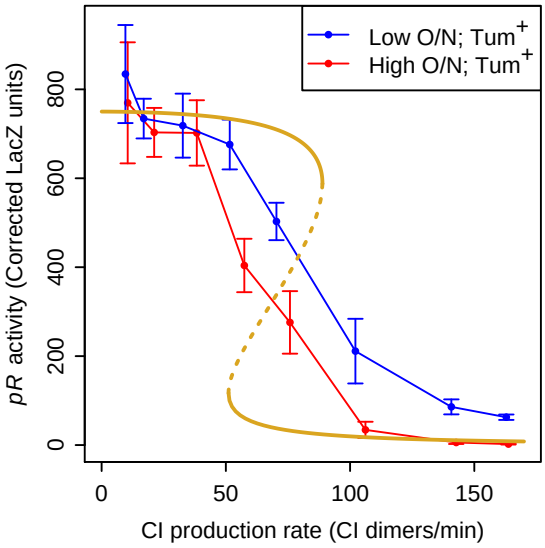


Figure 4.4: The steady-state model of bistability does not compare well with the long equilibration time assays. The low and high O/N Tum$^+$ LacZ assay curves from the WR-MC panel in Figure 4.3 are reproduced here overlaid with a steady-state curve derived from the Tum−CI MFL model. The model curve is essentially as previously depicted in Figure 2.7(b), but the model parameters were manually adjusted to improve consistency with the data.

dicator of the true SD, as judged by normal probability plots[7] of the weighted residuals from the sample means. This analysis was limited to the MFL strains with medium-copy *pR-tum* plasmids, since those with low-copy plasmids did not show persistent hysteresis, were hence less informative, and so will not be considered in the context of fitting parameters. As will be described below, alternative cost functions were employed to suit different purposes, but all functions involved collation of residuals across multiple data sets. In particular, the complete hysteresis assay — both Tum$^+$ and Tum$^-$ variants, each having low and high O/N data sets — was always scored as a unit, thus requiring calls to both repressor and MFL models in order to calculate the cost. In this way, the best parameters would maximise the fit to all relevant data sets simultaneously.

Ideally, the steady-state model of the Tum−CI MFL would be used for fitting the data since it has the least number of parameters. The data of choice would then be confined to the long equilibration time hysteresis assays, since the normal equilibration time assays are clearly further from steady state. A comparison of the WR-MC long equilibration time hysteresis assay data with a typical steady-state curve is shown in Figure 4.4. Whilst the model parameters in the figure were manually chosen, so do not represent an optimised fit of the data, the qualitative differences between the steady-state model and data are still apparent. The long equilibration time WR-MC MFL hysteresis assay (and also the ER-MC assay), bears a shape that appears 'collapsed' relative to that of the steady-state model. It is difficult to see how the bistable region of the steady-state model might fit inside the experimental hysteresis curves. Even reducing the vertical span of the unstable states (dotted line) to zero would not allow the steady-state model to encompass all of the apparently stabilised data points (recall that the low and high O/N curves of the WR-MC MFL strain remain separated at the highest rate of production for longer equilibration

[7] An assumption of the least-squares approach to regression is that residuals share the same variance (standard deviation) and fit a normal distribution, hence the use of normal probability plots.

times still).

So if the deterministic model as described is indeed suitable for describing the Tum−CI MFL hysteresis assays, the additional flexibility in curve shape afforded by time course simulations of the deterministic free species model may well be necessary to explain the observed collapsed hysteresis loops. Unfortunately, shifting to such a model necessitates additional parameters, specification of initial species concentrations, and choice of simulation times for each assay. However, in spite of the increased size of the parameter space, this model does still offer a couple of important advantages: (1) non-equilibrium behaviour can be captured, which means that both normal and long equilibration time data sets can be simultaneously fit; and (2) solutions calculated using this model are more robust to parameter variation than those of the steady-state model. One major disadvantage is the additional computation time such simulations require. With both normal and long equilibration time assays and also assays of the Tum^- controls, each of the ER-MC and WR-MC data sets contain 64 experimental factors for which residuals must be calculated. Since each of these experimental factors requires its own deterministic simulation to calculate the cost, and this cost may need to be recalculated up to 10000 times in the search algorithm, a number of code optimisations were incorporated into the model simulation routines to minimise computation time. The model implementation, fitting routines and optimisations are further described in Appendix C.

In order to limit the size of the parameter space, a few assumptions will be made to fix some parameters. However, since the aim here is to test whether the model can fit the data, the number of fixed parameters will be kept to a minimum to maximise model flexibility. The free species MFL ODEs (refer back to Equation (2.17) in Section 2.2.3) are defined in terms of 11 parameters, with an additional 2 parameters required to describe the evolution of the reporter gene product, β-gal. Of these 13 parameters, 5 can be eliminated since (1) the CI production rate (p_C) is set in the assay according to the normalisation described in Section 4.1.2, (2) the degradation rates of CI and β-gal (δ_C and δ_Z) are assumed dilution limited, so can be set equal to the dilution rate (γ), (3) the dissociation constant for the Tum−CI equilibrium (k_{-S}) only needs to be large enough that the Tum−CI reaction can occur on a different timescale, making the estimate from Chapter 2 sufficient, and (4) the degradation rate of the Tum−CI complex (δ_S) is left set to zero, since there is no evidence for such degradation and the parameter plays a similar role to ε_S in affecting curve shape. The remaining 8 parameters are the EC_{50}s and Hill coefficients of the Tum−CI and CI−pR equilibria, the production rates of Tum and β-gal from pR, the degradation rate of Tum and the dilution rate. To maximise the flexibility of the model all of these parameters will be allowed to vary, though some minor changes to the parameterisation will prove useful.

Since a biased random walk is employed to search the parameter space, the implicit correlation of production and degradation terms may result in futile cycles of variation in which these parameters change at cross purposes to each other. To minimise such redun-

dancy in the sampling, the maximum steady-state values of Tum and β-gal (M_T and M_Z respectively) are varied instead, and then for simulation, the respective production rates are derived from these steady-state values and the given degradation terms. Analogously, since the measurements of CI production rate as used in Section 4.1.2 were derived via the measured dilution rate, which is here allowed to vary, the level of induction of P_{lac} is fixed instead using the steady-state CI concentrations (M_C). Then for the simulation, all relevant CI production rates are derived from these steady-state concentrations. Finally, in order to limit the degradation rate of Tum so that it cannot be slower than the dilution rate, a parameter that specifies the ratio of Tum degradation to dilution is instead varied. Setting a lower limit of 1 on this scaling factor will prevent the Tum degradation rate from becoming slower than the dilution rate. This modified set of 8 parameters are the core determinants of MFL circuit behaviour, but there are still a number of additional parameters that are needed to describe the hysteresis assay.

When modelling hysteresis by a time-course simulation, the initial species concentrations and timing of the simulation can also impact the shape of the resulting curves. For the hysteresis assay described here, it is assumed that overnight growth of the cultures brings the circuit to steady-state at each end of the hysteresis curve. With that assumption, initial species concentrations can be derived using steady-state considerations. However, as highlighted in Section 4.2.2 to explain the stable separation of the WR-MC MFL curves at the highest CI production rate, the simulation parameters of a stationary phase culture are likely to differ from those of log phase cultures. To capture such a discrepancy without introducing too many additional parameters, two scaling factors are introduced that independently adjust the assumed CI production rates of the low and high O/N cultures. Initial species concentrations are then determined by equilibrating the circuit to steady-state according to the log phase parameters, but at these scaled CI production rates. With the initial effective CI production rates no longer restricted to those at the boundaries of the log phase assay, the simulation is able (with a suitable choice of parameters) to reproduce the observation of stable separation at the maximum level of induction that was seen for the WR-MC MFL. The use of these two initial scaling factors partly obviates the need for optimisation of the simulation times, since the scaling factors also affect how far the initial states are from their final equilibrium points. As such, simulation times for modelling of the normal and long equilibration data will be left set to the experimental averages.

4.3.2 Fitting the WR-MC data set alone

The deterministic model was tested first against the data of the WR-MC MFL. The fit to both normal and long equilibration time data sets was optimised simultaneously, since such time-dependent data would help to better constrain the kinetic parameters, notably the degradation and dilution rates. All 10 parameters as described were allowed to vary during the pseudo-random search, with parameter bounds set according to the confidence

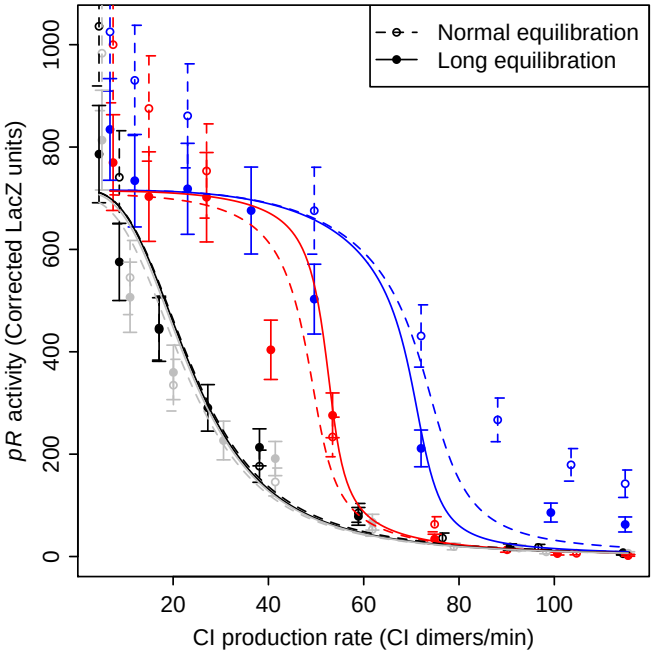


Figure 4.5: The deterministic Tum−CI MFL model cannot match all features of the experimental WR-MC MFL hysteresis assays. The model curves as shown were found via an extensive search of the parameter space using a pseudo-random search algorithm, and represent the best combined fit of the normal and long equilibration time hysteresis assay data for the WR-MC MFL strains (the original data is shown in Figures 4.1 and 4.3). Colour is used to distinguish Tum^+ and Tum^- strains and the initial O/N cultures (low or high) according to the previous figures. The error bars indicate standard deviations (SDs) used for weighting the sums of squared residuals during fitting; these are not the individual sample SDs, but were derived from a log-log regression of SD versus sample mean.

in the experimental parameter estimates. These bounds were iteratively enlarged according to some initial trials in which the parameters of best fit were close to the provided limits. The curves of best fit are shown overlaid on the data in Figure 4.5.

Whilst the long and normal equilibration time data sets do bear some notable inconsistencies with each other, the 'optimal' model fit clearly misses a number of important experimental data points. It captures the skewed shape of the collapsed experimental hysteresis loop to some extent, especially at the normal equilibration time point, but it fails to match the degree of skew seen experimentally. The most obvious deviation from the data is evident at high CI production rates, where the experimental data continues to show significant separation between low and high O/N curves, whilst the model curves quickly equilibrate to a single low *pR* activity monostable state. Hence, in spite of the introduction of stationary phase CI production scaling factors that can enable bistable separation at the highest CI production rate, the best fit of the data is obtained when the region of bistability sits within the region of largest separation between high and low O/N curves (around 50–80 CI dimers/min). The model cannot simultaneously capture both the separation of hysteresis at high CI production rates and the significant skew of the hysteresis data. It cannot even match the rapid change of the high CI, low O/N Tum^+ data between the normal and long equilibration time points. There are certainly discrepancies between these data sets, and this could be partly to blame for such insufficiencies, but with the obvious discrepancies being located around the lowest CI production rates with the highest *pR* activities, this cannot be the primary source of deviation from the model. So, as was also expected of the steady-state model, the deterministic model does not explain the unusual 'collapsed' shape of the hysteresis curves or the apparent existence of bistability at

the highest CI production rate.

The Tum$^-$ control data shows fewer discrepancies between normal and long equilibration time data, and supports the earlier statement that the biggest discrepancies between the two data sets are at the lowest CI production rates. This data describes a much simpler gene network primarily consisting of the effect of CI repression at *pR*, but even so, the fit of the model control curves to this data set were not as good as expected. Firstly, about the half maximal activity point, the model curves overestimate the activity at lower CI production rates and underestimate it at higher CI production rates. Secondly, maximum *pR* activity should occur in the absence of CI, but the model curve seems to approach a maximum that is well below the mean activities of the (non-zero) minimum CI production rate data points. These deviations are reflected in the optimised parameters: the Hill coefficient for CI repression (H_R) is estimated unusually high at around 3, and the maximum *pR* activity (M_Z) is grossly underestimated at around 715 LacZ units. Fits of the control curves by themselves produced more plausible models of CI repression of *pR* (data not shown), so the apparently poor fit shown in Figure 4.5 is presumed to be a consequence of the additional constraints imposed by simultaneously fitting the MFL data. This further highlights the deficits in the MFL model, since the poorly fitting MFL data also compromises the fit of the control curves.

4.3.3 Fitting the combined data sets

Whilst the WR-MC MFL strain produced the best separation between high and low O/N hysteresis curves, modelling of that strain was complicated by the fact that the hysteresis loop was incomplete. In contrast, the ER-MC MFL strain, which has the same theoretical steady-state behaviour as the WR-MC strain, produced a complete hysteresis loop since it can access higher CI production rates during log phase growth. With the additional information provided by the ER-MC strain, it may be possible to better constrain the search of the parameter space and find more likely fits of the data. However, to make use of this additional data it becomes necessary to draw a more quantitative comparison between the ER-MC and WR-MC data sets.

When close to steady-state, the ER-MC and WR-MC Tum$^-$ control strains should give similar *pR* activities at points where the production rate of CI is induced to the same level. However, by comparing the x-axes of the experimental curves in Figure 4.3, it can clearly be seen that the CI production rates assigned to the eRBS strains conflict with those assigned to the wRBS strains. This was not an unexpected discrepancy since the scaling of CI production rates was based on experimental measurements of steady-state CI concentrations, as described in Section 4.1.2, and those measurements were associated with high levels of uncertainty. Whilst such production rate measurements could be introduced as additional variable model parameters, it is preferable that they remain fixed when searching the parameter space, since these steady-state CI concentrations set the positions of the

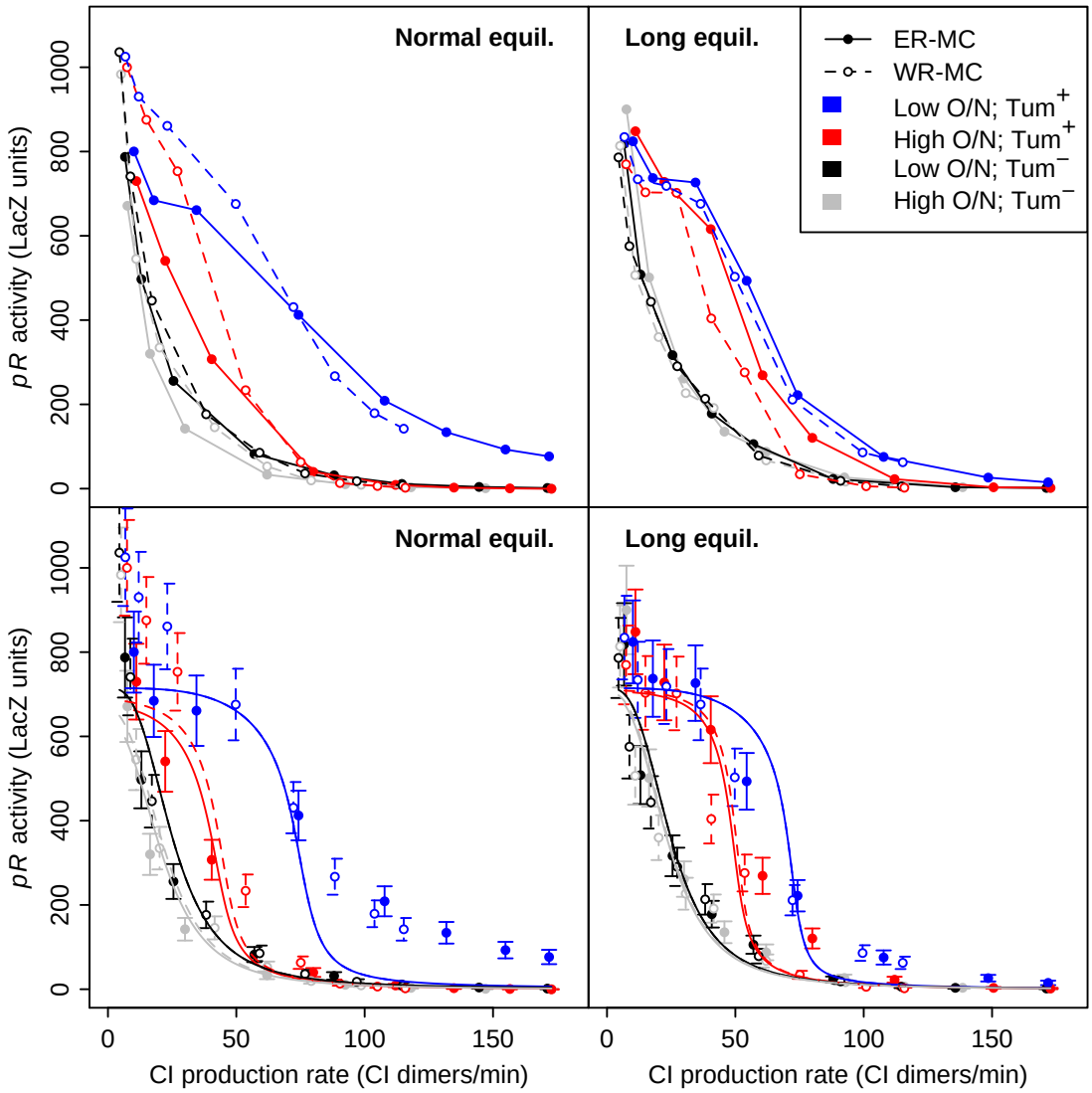


Figure 4.6: The whole-population WR-MC and ER-MC data sets can be matched up, but the additional data does not improve the model fit. The upper panels show the combined data of all hysteresis assays for the WR-MC and ER-MC strains at both normal and long equilibration (equil.) time points after scaling the CI production rates ER-MC data set to align that data with the WR-MC data set. The alignment was determined by optimising a single parameter, effectively the relative increase in strength of the enhanced RBS (ER) over the wild-type RBS (WR), to fit a reference model (defined by parameters optimised to fit the WR-MC data) to the ER-MC data set. The lower panels show the same data as the upper panel, but overlaid instead with model curves that were obtained by optimising 10 parameters (as described in Section 4.3.1) using a pseudo-random search algorithm to explore the parameter space. Error bars indicate the SDs used for weighting the sums of squared residuals as described in Figure 4.5.

data points and thus serve as key reference measures. By leaving them fixed, any errors in the scale of such parameters are instead compensated for during fitting by correlated adjustments to other parameters such as the EC_{50} of CI$-pR$ binding (ε_R), or the dilution rate (γ). Since the measurements of CI production rate appear to be substantially different for each RBS, a single parameter set could not simultaneously describe both the WR-MC and ER-MC data. A way to reconcile the ER-MC data set to the WR-MC data set is needed.

To obtain a consistent combined data set, the CI production rates of the ER-MC data set were scaled to align that data with the WR-MC data. This was achieved by optimising a scaling factor for CI production rates to fit the ER-MC data set with a reference model, derived from the optimised parameters of the WR-MC data set. Since only this scaling factor was optimised, a shorter non-random optimisation algorithm was employed. Application of this optimised scaling factor (with a final value of 0.57) to the ER-MC data set produces a remarkably good overlap between the ER-MC and WR-MC data sets, as shown in the upper panels of Figure 4.6. This is particularly evident for the Tum^- control curves, but considering the sizes of the confidence limits on the MFL data points shown in Figures 4.1 and 4.3, there also seems to be good agreement between the ER-MC and WR-MC MFL curves. In spite of the scaling, the adjusted eRBS CI steady-state concentrations still fall within the experimental limits. The measurements of CI steady-state concentration reported in Section 3.5.4 are listed in Table 4.1 along with the corresponding scaled CI steady-state concentration for the eRBS. Whilst close to the lower limit, the scaled value does lie within the confidence limits of the experimental measurement. Thus, it is likely that the eRBS makes a smaller difference to the CI production rate than initially thought.

With the data sets aligned in this way, the 8 core model parameters and 2 O/N scaling factors were optimised using the pseudo-random search algorithm as described previously for the WR-MC data set, but now for a combined cost function that would simultaneously fit both the WR-MC and ER-MC data sets. As seen in the lower panels of Figure 4.6, the optimum set of parameters produced model curves not unlike those found by fitting the WR-MC data set alone. The gentle descent of the low O/N MFL curve remains poorly fit, so like before, the bistable region defined by the model curves is not large enough to include the seemingly stable separation seen at the highest CI production rates of the WR-MC strain. It is now quite clear that the deterministic model cannot account for the collapse of the hysteresis curves.

The inadequacy of this model for describing the whole-population data sets is further compounded by some of the extreme parameter values that were needed to produce even poorly-fitting model curves. The optimised set of parameters is listed in Table 4.1, and whilst some of the parameters fall near the estimates obtained experimentally or derived from the literature, notably the unrepressed steady-state level of Tum (M_T), the majority did not. Of particular concern were the high Hill coefficient for the CI$-pR$ equilibrium, the low Hill coefficient for the Tum$-$CI equilibrium and the long cell doubling time and Tum

Table 4.1: Comparing parameters measured experimentally or obtained from the literature with those determined by fitting the deterministic model to the normal and long equilibration time ER-MC and WR-MC hysteresis assays. The upper section lists the parameters used to set CI production rates; the lower section lists the 10 parameters that were optimised to obtain the curves in Figure 4.6. Unless otherwise indicated, limits specify 95% confidence intervals.

Parameter	Experimental*	Literature	Model fit**
$C_{\text{tot}}^{\text{wRBS}}$ (CI dimers/cell)†	$(7 \pm 2) \times 10^3$	$(3 \pm 1) \times 10^3$ ‡	–
$C_{\text{tot}}^{\text{eRBS}}$ (CI dimers/cell)†	$(1.7 \pm 0.8) \times 10^4$	–	$(9.9 \pm 0.2) \times 10^3$
Low O/N scaling parameter	–	–	0.16 ± 0.02
High O/N scaling parameter	–	–	4.9 ± 0.1
ε_R (CI dimers/cell)	–	$(2.5 \pm 0.8) \times 10^2$ §	$(1.43 \pm 0.09) \times 10^3$
H_R	–	2.2 ± 0.2 §	3.0 ± 0.1
ε_S (Tum monomers/cell)	–	$(1.28 \pm 0.04) \times 10^3$ ¶	$(7 \pm 3) \times 10^3$
H_S	–	4.6 ± 0.6 ¶	1.00 ± 0.02
M_Z (LacZ units)	–	800–1200 #	714 ± 61
Doubling time (min.)	28.2 ± 0.2 ††	–	40 ± 3
Tum half life (min.)	2.7 ± 0.2	–	39 ± 4
M_T (Tum monomers/cell)	$(2.5 \pm 0.7) \times 10^4$	–	$(3 \pm 1) \times 10^4$

* Parameters determined experimentally in this thesis.
** Parameters determined by fitting hysteresis data with the deterministic model.
† Steady-state levels of total CI dimers measured for the wild-type RBS (wRBS) or enhanced RBS (eRBS) with induction by 300 μM IPTG.
‡ Extrapolated from Western blotting data of Dodd and Egan [2002] as described in Figure 3.11, Section 3.3.
§ Parameters as fit by Dodd et al. [2007b]; limits indicate 'acceptable ranges'.
¶ Obtained by fitting *in vitro* gel shift data of Shearwin et al. [1998] to the model described in Section 2.2.2; ε_S was further converted to a number of monomers per cell using the scaling factor from Section 3.4.4.
# Range estimated from LacZ assays of various MFL-like CI^- control strains in Pocock [2007].
†† The mean doubling time of all MFL strains (Tum^+ and Tum^-) with a medium copy *pR-tum* plasmid.

half life. All of those parameters encroached on the chosen boundaries for the parameter space, so marked significant departures from the expected values. The high Hill coefficient of the CI$-$*pR* equilibrium is again responsible for the poor fit of the control curves. So, considering that the fitted model also argues for no Tum$-$CI cooperativity, the indication is that a fundamentally different mechanism would be required to explain the action of Tum on CI *in vivo*. Alternatively, some other perturbation of the data may have forced these parameters into explaining an unrelated phenomenon.

Though deterministic models have often been successfully employed to describe the behaviour of genetic circuits [Gardner et al., 2000; Elowitz and Leibler, 2000; Karlebach and Shamir, 2008], gene networks are intrinsically noisy [Raj and van Oudenaarden, 2008] and can exhibit phenotypes that are products of such stochasticity [Cağatay et al., 2009; To and Maheshri, 2010]. The 'collapse' of hysteresis in the MFL strain could well be ex-

plained as a stochastic and not a deterministic phenomenon: if the *pR* activity recorded at each IPTG concentration is the average over a population of cells with heterogeneous final CI concentrations, the sharp bifurcation boundaries of the deterministic model could be significantly blurred. Conceivably, this source of noise alone could be the origin of the skewed hysteresis curves if the stochastic expression of CI were noisy enough. However, it is likely that other sources of intrinsic noise also arise in this gene network. Such explanations for the collapse of hysteresis will be explored in subsequent chapters.

4.4 Chapter summary

The experimental hysteresis assays of the Tum−CI MFL strains that were first reported in Chapter 3 did not compare well with the predictions of the deterministic model. Correcting for the dependence of LacZ units on OD_{600} and normalising the culture IPTG concentrations to corresponding CI production rates, helped to allay concerns that this could have had an experimental origin. Nonetheless, these corrections did still increase the apparent separation between the hysteresis curves and also facilitated cursory comparisons with the model. These considerations brought into attention the possibility for non-equilibrium behaviour to obscure such comparisons. This motivated modifications of the hysteresis assay that extended the time of equilibration, and a set of long equilibration time hysteresis curves were measured. These new curves revealed a collapsing hysteresis loop, showing no bistability for the MFL strains with low-copy *pR-tum* plasmids, and only weak bistability for the MFL strains with medium-copy *pR-tum* plasmids. The shapes of the hysteresis curves were unexpected, so attempts were made to fit the deterministic model to the combined data set of all medium-copy *pR-tum* strains. After an extensive search of parameter space, it was decided that whilst the deterministic model could approximate the data sets, it was unable to account for a number of important features. These unexplained perturbations will be addressed in the following chapters.

5

Hysteresis of the Tum−CI MFL is obscured by stochastic switching between two semi-stable states

Hysteresis assays of the Tum−CI MFL by LacZ assay provided evidence for bistability that was based on the observation of history-dependent curves. The limited persistence of these hysteresis curves, however, left in question the value of the assay for assessing regions of bistability in the MFL. Parameter variation in the model alone could not account for the observed skew of the hysteresis cycle, and it was instead hypothesised that the anticipated boundaries had been blurred by sources of noise that were invisible to the population averages obtained by LacZ assay. With sufficient cell-to-cell variation, averaging of the MFL reporter activity could result in a biased measure relative to the deterministic model. Cell-to-cell variation is not without precedent: noise in protein abundance is generated both intrinsically through gene expression and extrinsically through variation in other cellular factors [Elowitz et al., 2002]. Stochastic variation in protein abundance can give rise to alternative phenotypes that are dependent on noise characteristics [Maamar et al., 2007; Cağatay et al., 2009], can extend effective bifurcation boundaries relative to deterministic predictions [Kepler and Elston, 2001], or can even produce bimodal behaviour in networks without deterministic bistability [To and Maheshri, 2010]. In order to determine whether such stochastic effects might result in the limited bistability observed by LacZ assay, a measurement technique is required that can record reporter gene levels in individual cells. Whilst single-cell gene activity measurements can indeed be made using a *lacZ* reporter gene [Cai et al., 2006], it is far simpler and more common to make use of fluorescent reporters, measuring cell fluorescence by microscopy [Becskei and Serrano, 2000; Elowitz et al., 2002; Yu et al., 2006] or flow cytometry [Gardner et al., 2000; Isaacs et al., 2003].

In this chapter, robust automated protocols are developed for measuring and analysing Tum−CI MFL cell samples from hysteresis assays by flow cytometry. The chapter begins with a description of the flow cytometric method and details data preparation methods that improve the relationship between fluorescence data and internal cell state. A preliminary look at the resulting curated fluorescence distributions suggests that samples in the putative regions of bistability are in fact mixtures of two fluorescence populations. A

definitive analysis is unfortunately obscured by poor separation between the populations, which then prompts a more rigorous quantitation of the reporter distributions. An empirically constrained mixture model analysis provides high-confidence confirmation for the presence of two cell sub-populations in many of the samples. Using summary statistics to describe each fitted sub-population, a new picture of hysteresis emerges. The long equilibration time assays of each of the Tum−CI MFL strains now clearly reveal a region supporting two semi-stable states that compares favourably with the steady-state model of the MFL. A similar analysis for single-cell measurements of the ER-MC MFL for different equilibration shows that switching between these semi-stable states occurs on a timescale similar to the length of the hysteresis assay.

5.1 Relating single-cell and whole-population measures of promoter activity

Given the typical magnitudes of gene expression noise seen in *E. coli* [Taniguchi et al., 2010], single-cell measurements need to be obtained for relatively large populations of cells in order to accurately sample the distribution of activities. Using flow cytometry, the fluorescence of many thousands of cells can be measured for a single culture; this set of fluorescence measurements is a representative sample of the 'true' fluorescence distribution. In the hysteresis assay, such distributions will be sampled over multiple experimental conditions, and reliable automated protocols that maximise the relationship of fluorescence intensity to protein levels form an important part of the analysis. This section describes methods for filtering and normalising single-cell measurements made by flow cytometry. The average intensities of the curated data sets are used to confirm that these methods produce results that are consistent with those obtained by LacZ assay.

5.1.1 Assaying gene circuit hysteresis by flow cytometry

To maximise consistency with the previous assays, methods for growing and preparing cells for measurement by flow cytometry were kept as close as possible to those developed for the LacZ assay. As described in Section 3.1.2, all of the Tum−CI MFL strains used in this thesis were designed with a dual reporting system in which separate *pR-lacZ* and *pR-gfp* reporter modules allow for measurement of the same culture by either LacZ assay or flow cytometry. Hence, the protocols for hysteretic growth of the MFL and *tum*$^-$ control strains remain unchanged except for the final assay step. Whereas in the LacZ assay, cells in the culture are lysed once the chosen OD_{600} is reached, in the flow cytometric assay, cells at the chosen OD_{600} are instead resuspended in Phosphate-Buffered Saline (PBS) prior to measurement in the flow cytometer. This means that the different assay types mentioned in Chapter 4 (i.e., short, normal and long equilibration time assays) are all still applicable in the single-cell experiments. By taking single-cell measurements, flow

cytometry removes the direct dependence on the OD_{600} that was needed for calculating LacZ units. Nonetheless, similar OD_{600}s are maintained to ensure that cells are measured at comparable growth phases and equilibration times.

Measurements obtained using the flow cytometer provide a much more comprehensive picture of the state of the culture than the LacZ assay. Most importantly, the flow cytometer can record a large number of single-cell measurements. By making use of a specialised fluidics system, a flow cytometer can focus a flowing cell suspension so that cells pass through a laser beam one at a time. The laser beam excites any fluorophores present in a given cell, which then fluoresce at wavelengths longer than that of the laser. Different fluorophores emit at different wavelengths, and an array of detectors and optical filters located at 90° to the beam can be used to quantitate the fluorescence emitted by different fluorophores simultaneously. As described in Section 3.1.2, the folding reporter GFP was chosen since its fluorophore maximises overlap of the excitation and emission spectra with the laser light wavelength and filters of the available flow cytometer.

The flow cytometer also measures laser light that gets scattered by the cells. The forward scatter (FSC) detector (in the path of the beam) measures small-angle scattering of the laser beam due to a given cell, and is related to cell volume [Koch et al., 1996]. The side scatter (SSC) is a measure of the laser light that is scattered 90° to the beam, and is affected by the internal complexity/granularity of each given cell. As a cell traverses the beam, each of these detectors sees a pulse of light, and the intensity is measured as both a pulse area and peak height. So, in a configuration with just one fluorescence detector active (i.e., the configuration for the Tum−CI MFL), the flow cytometer returns 7 different channels of data (including the relative measurement time of each event). However, unless otherwise specified, standard practice is followed throughout this thesis by using only the FSC peak height channel (FSC-H), the SSC pulse area channel (SSC-A), and fluorescence pulse area channel.

In a manageable time frame (considering the number of cultures involved in a typical hysteresis assay), such measurements can be obtained for between 30,000 and 100,000 cells for each culture[1]. Sources of experimental variation that are extrinsic to a given culture, such as technical variation, are not captured in the cell-to-cell variation of a single culture. However, given the size of the hysteresis assay and the extra time required for flow cytometric measurements, such technical repeats were not made. In the LacZ assay, technical repeats were especially helpful, given the close dependence of the measurement on culture OD_{600}; this source of variation is less influential on the flow cytometry results, improving consistency between technical repeats, but ideally such technical repeats would be made. In their raw form, these large data sets contain factors that obscure an accurate analysis, so some curation must be performed before interpreting the results. This includes data transformation, filtering out non-cellular debris, and normalising the fluorescence according to

[1]Note that LacZ assay measurements capture the average activity of around 2×10^6 cells.

cell morphology; each of these factors will be addressed in the coming sections.

5.1.2 Choosing an appropriate data transformation

Flow cytometry data has commonly been visualised on a logarithmic scale, since this tends to produce intensity distributions that are more normally distributed, and facilitates the visualisation of data that can be spread over several orders of magnitude. For samples with low or no fluorescence, instrumental background correction can result in measurements that are less than or equal to zero, and these cannot, therefore, be displayed on a logarithmic scale. The 'Logicle' display method [Herzenberg et al., 2006; Parks et al., 2006] overcomes this deficiency by taking a biexponential transformation of the data that is close to logarithmic for large intensities, but approximately linear for measurements around zero. Using this method, zero and negative measurements can still be displayed and analysed. Parameters for the logicle transformation function are chosen according to the protocol specified in Parks et al. [2006] for a 4.5 decade display, and the transformation implemented using the flow cytometry packages from the Bioconductor collection of R packages for the biosciences [Gentleman et al., 2004; Ellis et al., 2009a]. As described in Parks et al. [2006], the fifth percentile of the negative values is chosen as the negative range reference value for determining the width parameter. To maintain a consistent transformation over an entire assay (to within each dimension/channel), these parameters are derived using the aggregated negative values of all samples in that assay, and determine an assay-specific Logicle transformation. A number of subsequent analyses make use of the Logicle-transformed fluorescence intensities since this simplifies the choice of statistical distributions; where transformed data is specified, the assay-specific Logicle transformation is assumed. Whilst FSC and SSC channels typically contain no negative values, where these are more appropriately treated using a logarithmic scale, the Logicle transformation is still applied for consistency and also so that no quantitative information is lost in transformation.

5.1.3 An automated filter for selecting cell populations

An unavoidable component of any cell suspension is a level of contamination resulting from dust particles, dead cells, and other non-cellular debris. The FSC and SSC channels can be used to filter out most of these unwanted data points, since healthy cells tend to cluster according to size and subcellular composition. In order to automate this process, the clustering functions of the `flowClust` package [Lo et al., 2009] were used to select the primary population of cells using an elliptical gating strategy. The ellipse is chosen so that 95% of the data points are included. Whilst such a liberal inclusion rate is not typical, a morphology normalisation strategy will be later applied in order to maximise cell homogeneity, removing the disadvantages of retaining such heterogeneity. An example application of a 95% elliptical filter is shown in Figure 5.1, where it can be seen that the

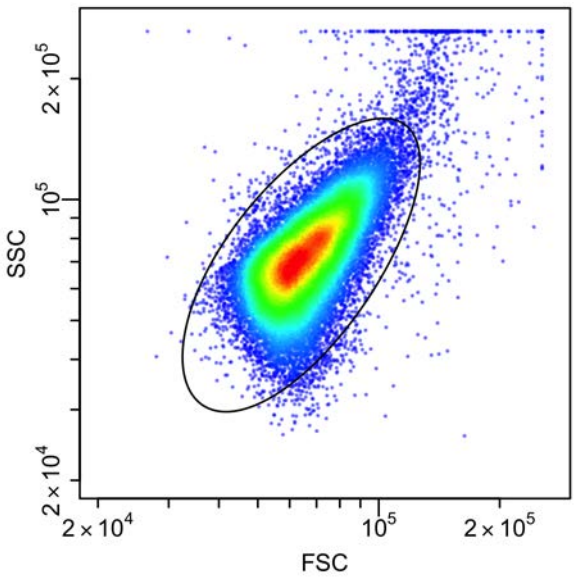


Figure 5.1: The main bacterial cell population is easily identified using the forward scatter (FSC) and side scatter (SSC) intensities. Shown plotted are the SSC intensities (recorded as peak area) versus the FSC intensities (recorded as peak height) as measured by flow cytometry for a sample of ER-MC MFL cells grown according to the normal equilibration time assay with 60 µM IPTG, starting from a low O/N culture. The intensities are shown plotted using a Logicle display, which is effectively logarithmic at these high intensities. To assist visualisation, points are coloured by density. The black ellipse shows the 95% cutoff as determined by a clustering algorithm for a single cell population. Out of 31,770 events, 30,619 lie within the ellipse and are kept for further analysis.

majority of events recorded by the flow cytometer cluster close to the primary cell population. This is highly representative of all FSC/SSC plots regardless of strain or assay. After filtering, the remaining events are presumed to be the cells of interest.

The efficacy of such filtering can be verified in part by comparing how the resulting fluorescence distributions compare with the activity expected for the *pR* promoter. The upper panels of Figure 5.2 show histograms of the recorded fluorescence intensity for selected cell samples of a hysteresis assay. At the highest induction level (300 µM IPTG), the *pR* promoter is expected to be close to fully repressed, and the cell population reflects this with a relatively low fluorescence that represents the background fluorescence (or autofluorescence) of the cells. In contrast, at the lowest induction level (0 µM IPTG), the *pR* promoter is expected to be close to fully active, and the significantly higher fluorescence of that population reflects this. Very few events lie outside this high fluorescence population (less than 0.5% of the events have intensities outside 2×10^3–2.5×10^4), indicating that the vast majority are behaving as expected for the bacterial cell population.

The histogram for the intermediate induction level (60 µM IPTG) reveals that populations in the putative region of bistability have a more complicated probability structure. Nonetheless, by comparing this sample with the others, it seems feasible to suggest that this sample might be a mixture of low and high fluorescence populations.

5.1.4 Morphology normalisation refines the distribution of fluorescence

The apparent mixing of high and low fluorescence populations in the intermediate induction sample of Figure 5.2 highlights the poor separation between the minimum (300 µM IPTG) and maximum (0 µM IPTG) fluorescence populations. Whilst the peaks in the intermediate sample do not precisely overlap with the minimum and maximum peaks, an inspection of the peak widths (using the axis labels as a reference) makes it clear that a mix-

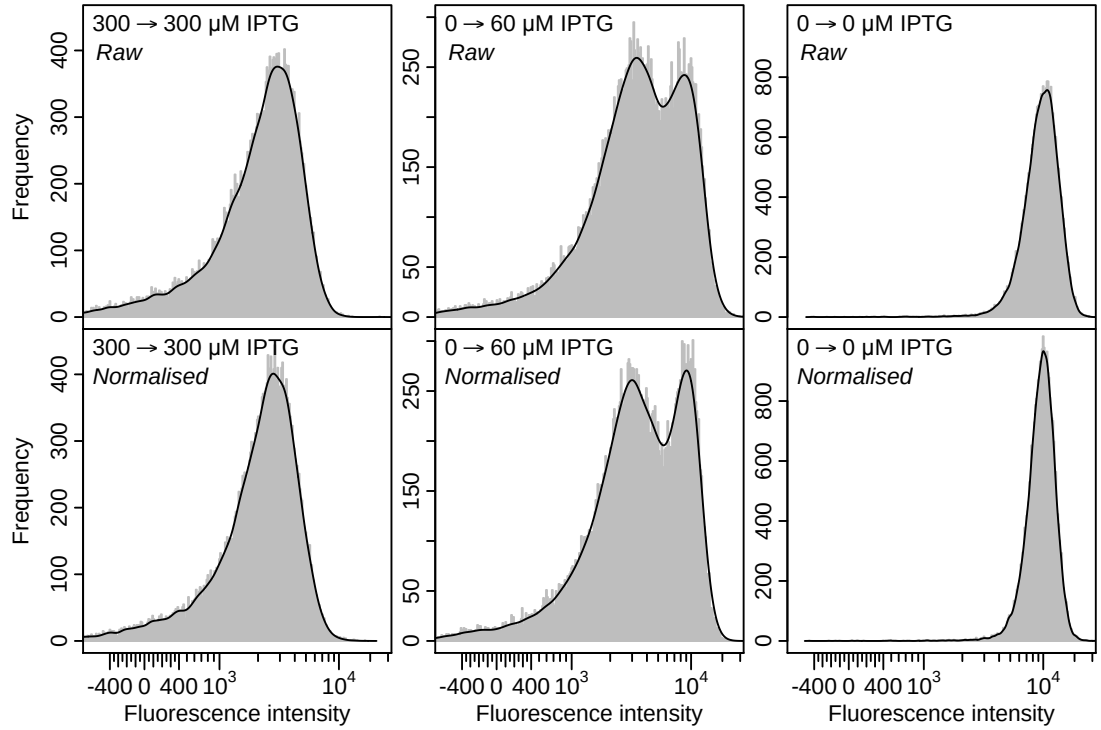


Figure 5.2: The resolution between low and high fluorescence populations is poor. Shown are histograms of fluorescence intensity for cell populations of the ER-MC MFL strain grown according to the normal equilibration time hysteresis assays for the initial and final IPTG concentrations as indicated. Included in the histograms are only those events which passed the filter as described in Figure 5.1. The histograms are displayed on a Logicle scale (see Section 5.1.2); the approximate limits of the linear region about 0 are indicated. Histograms in the upper panels are of the raw fluorescence intensity, whilst those in the lower panels are of the fluorescence intensity after normalisation according to morphology. The solid black line in each histogram is the kernel density estimate.

ture of low and high fluorescent populations would nevertheless suffer a significant level of overlap. This poor resolution prompts a closer look at variables which may obscure an accurate measure of *pR* activity. Such deficiencies in resolution may merely reflect the intrinsic level of noise produced by the genetic network, but other sources of variation/error are also likely. These could include instrumental measurement errors, poor fluorescence sensitivity over the cellular autofluorescence background, or variations in protein number resulting from cell cycle-dependent differences.

Without resorting to changes in the strain or experimental procedure, the simplest source of additional variation to address is the latter. That is, the natural correlation of protein number with cell size: a cell that has just undergone division is expected to have roughly half the number of proteins (and hence half the fluorescence) of a cell just before division. Correcting for such an effect makes sense in terms of the deterministic model as well, since state variables are most easily treated as constant-volume concentrations, where the concentration of each species relative to the genome copy-number is the relevant quantity. For data obtained by flow cytometry, normalisation by cell morphology is a real possibility since the FSC and SSC intensities are indicators of cell size and complex-

ity. Such normalisation has seen successful application in *S. cerevisiae* [Knijnenburg et al., 2011], and is likely to have a similar, but perhaps less pronounced effect in *E. coli*.

Here, morphology normalisation is applied essentially as described by Knijnenburg et al. [2011], though using custom R scripts instead of Matlab scripts to perform the analysis (see Appendix B). Briefly, the normalisation protocol estimates for each sample the general trend of fluorescence intensity with both FSC and SSC, makes relative corrections for this trend within the sample, and finally restores absolute fluorescence intensities according to a model FSC/SSC density. The general trend of fluorescence intensity is characterised as a strictly increasing regression surface, defined over FSC and SSC space (with terms as specified in the paper), that is fit to the (filtered) data for each sample[2]. The residuals from this regression surface are kept as the corrected deviation of the population from the fluorescence mean. The mean fluorescence is then calculated from the surface as the expected fluorescence over a representative FSC/SSC density. The choice of this representative density will be returned to later.

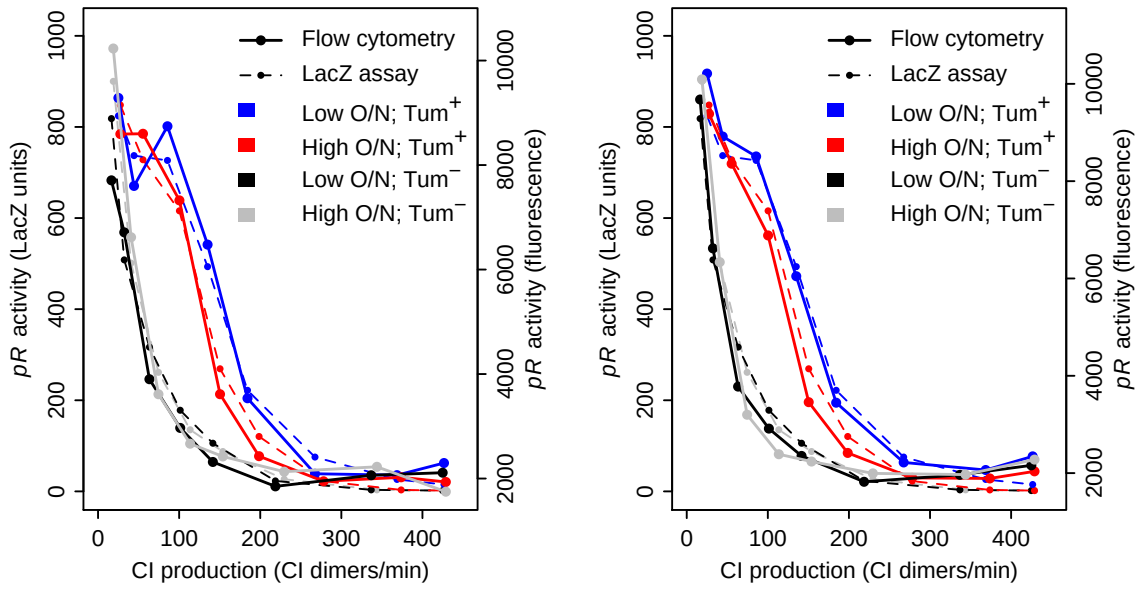
Application of the morphology normalisation to Tum−CI MFL cell samples provides a marginal improvement in resolution between high and low fluorescence populations. Referring back to Figure 5.2, the effect of morphology normalisation can be seen by comparing selected samples before (upper panels) and after (lower panels) normalisation[3]. There is a small but definite increase in resolution that is especially evident as a deeper trough between the putative high and low fluorescence peaks of the intermediate induction sample, but is also evident as a sharpening of the peaks in the other two samples. Whilst these differences may appear inconsequential at this point, the normalisation deconvolutes hidden variables from the samples and hence better reveals the underlying model-related probability distributions, as will be confirmed later.

As a validation of the normalisation method, it is useful to consider the mean fluorescence intensity of each sample. The mean fluorescence intensity should in theory show a linear relationship with the equivalent results obtained by LacZ assay. Note that it is the untransformed fluorescence intensity that is proportional to the level of fluorescent protein (promoter activity), and not the Logicle-transformed intensity. Morphology normalisation is calculated using the Logicle-transformed data, so whenever population statistics are calculated in this thesis, the inverse Logicle transformation is always applied to the normalised data prior to deriving that statistic. In Figure 5.3(a), the mean fluorescence of morphology normalised data for a long equilibration time assay of the ER-MC strain is shown overlaid with LacZ assay results for comparison[4]. Since the units of each measurement

[2] To best match the method of Knijnenburg et al. [2011], the regressors are the untransformed FSC and SSC channels, whilst the response variable (fluorescence intensity) is treated logarithmically; here, though, the Logicle transformation is used in preference to the log transformation.

[3] Note that the normalisation applied in Figure 5.2 makes use of a minor conditional amendment to the protocol that will be introduced later.

[4] Note that to maximise comparability, the cell suspensions for these flow cytometry measurements were sampled directly from the same cultures as one of the repeats of the long equilibration time ER-MC LacZ



(a) Normalised using an average cell density

(b) Normalised using sample-specific densities

Figure 5.3: The mean fluorescence of the cell populations overlaps well with the mean *pR* activity measured by LacZ assay to within an offset and scaling factor. The population mean fluorescence intensities calculated for ER-MC MFL strains grown according to the long equilibration time hysteresis assay and measured using flow cytometry, are shown plotted against the normalised CI production rate, and are displayed with the equivalent results obtained by LacZ assay for comparison. Data from the LacZ assay is plotted according to the left axis; data from the flow cytometry assay is plotted according to the right axis, which has been scaled to maximise overlap between the two different data sets. The flow cytometry data is displayed for two variations on the morphology normalisation protocol: **(a)** normalisation where the average cell density is uniformly applied to determine the expected fluorescence for each sample, or **(b)** normalisation where the expected fluorescence for each sample is determined using a sample-specific cell density.

technique have different scales, and since fluorescence measurements have a non-zero offset due to cellular autofluorescence, linear regression was applied to maximise overlap of the two data sets by a scaling of the axes. On the whole, the LacZ assay sample means and flow cytometry population means of Figure 5.3(a) show surprisingly good overlap, especially considering that the mean activities obtained by flow cytometry do not factor in any technical (i.e., sample-to-sample) variation.

The representative cell density in FSC/SSC space specified by Knijnenburg et al. [2011] worked well for morphology normalisation of most samples. This is evident from the overlap of the mean fluorescence activities with LacZ activities as shown in Figure 5.3(a). However, the high intensity samples showed some biases that were found to be due to the choice of representative cell density. By modifying the protocol to use sample-specific cell densities, overlap for the high intensity samples was improved, as can be seen in Figure 5.3(b). This revised protocol was used instead of the original for these flow cytometry samples. A set of high quality assays that will be introduced later in Section 5.4 did not

assays reported in Section 4.2.2 and, as such, the LacZ assay data is taken directly from the ER-MC panel of Figure 4.3.

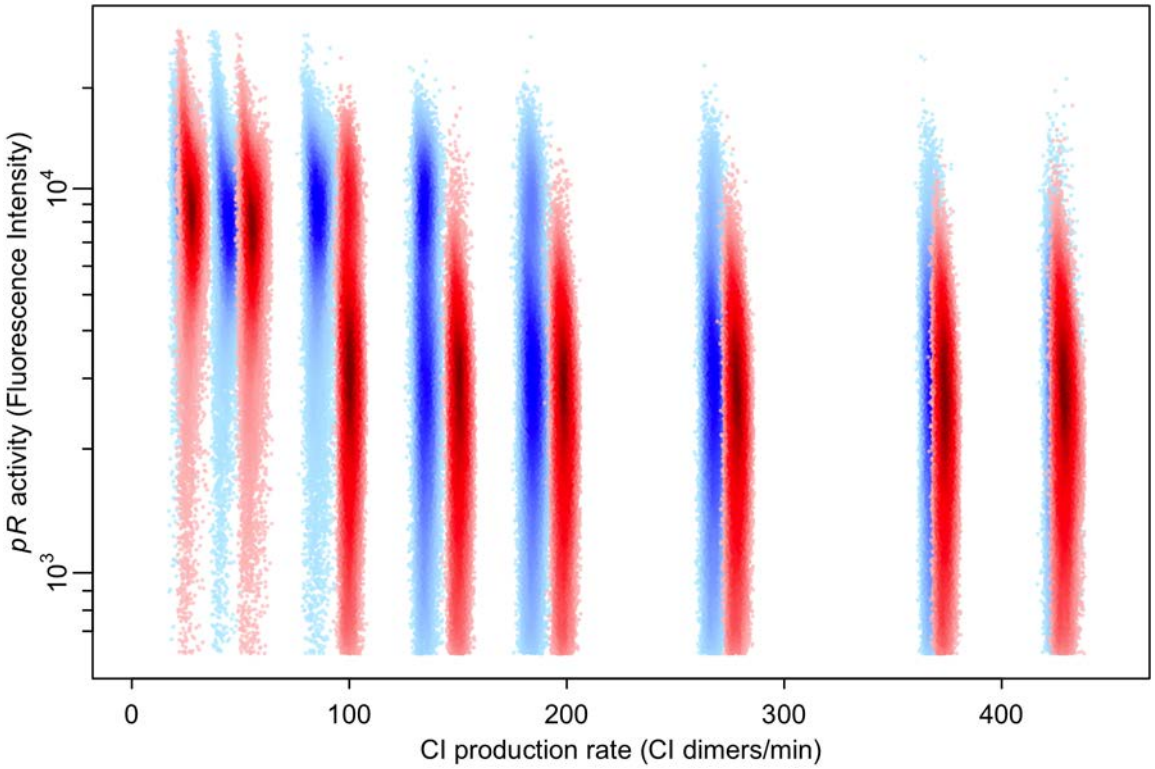


Figure 5.4: Viewed as cell populations, hysteresis in the Tum−CI MFL is manifest as a history-dependent broadening of fluorescence. Scatter plots of fluorescence versus FSC are plotted (using a Logicle display) for cell samples from a normal equilibration time hysteresis assay of the ER-MC MFL strain. The FSC has been offset and scaled to centre each plot at the rate of CI production that it was measured for. Cell populations starting from the low O/N are shown in blue and those from the high O/N are in red; points are shaded according to density.

suffer such errors, and in those cases the original protocol was preferred. More details of these two methods and the rationale for revising the protocol can be found in Appendix B.

5.1.5 The curated data is suggestive of population mixing

In spite of transformation, filtering and normalisation, the curated fluorescence data remains difficult to interpret due to the low resolution between the maximum and minimum intensity distributions. A comparison of the curated fluorescence distributions across an entire assay is shown in Figure 5.4 for a normal equilibration time hysteresis assay of the ER-MC MFL Tum$^+$ strain. To aid visualisation, the data for each sample has been separated according to FSC and shaded according to density. The low resolution limits interpretability, but a couple of interesting features are nonetheless apparent in the data without further analysis.

Firstly, asymmetry between the two alternative hysteresis curves is most obviously seen as a difference in distribution spread. The greatest spread in fluorescence intensity for the low O/N samples occurs at a higher production rate (around 130 CI dimers/min) than for the high O/N samples (somewhere between 50–100 CI dimers/min). This history-

dependent spreading of the fluorescence distributions cannot be seen using population averages alone (i.e., by LacZ assay), but represents a likely origin for the lack of sharp boundaries in the whole-population results.

Secondly, the presence of two sub-populations, as was suggested for the intermediate induction sample in Figure 5.2, appears to be consistently reproduced for a number of samples in the assay. By focussing on the regions of highest intensity (darkest shading), there appear to be two alternative populations of cells that share common average intensities, independent of initial condition or CI production rate, but that vary in relative abundance from sample to sample. The centres of these putative sub-populations line up with the centres of the minimum and maximum intensity distributions that bookend the hysteresis assay.

For a strongly bistable system, one would expect that cell populations set in monostable regions of the hysteresis curve would produce unimodal populations in the bistable region until the bifurcation threshold is reached. However, if the system were only weakly bistable, so that noise in the circuit could cause stochastic transitions between the two zones of stability, one might instead expect a mixture of two populations — a bimodal population — with some number of cells potentially 'in transition' from one state to the next.

5.2 MFL samples within the bistable region are a mixture of two cell populations

By inspecting the histograms and density maps of cell population fluorescence shown in the previous section, it seems likely that samples within the putative bistable region consist of two sub-populations. However, especially given the poor resolution between the two populations, a suitable statistical description of such populations will allow quantitative assessment and treatment of the sub-populations, facilitating interpretation of the data in light of the model. In this section, the skew-*t* distribution is demonstrated to fit the unimodal control populations very well. Then, using the fits of the controls as training sets, it becomes possible to define a sub-family of skew-*t* distributions that match the unimodal control samples. By fitting two-component mixtures of these constrained skew-*t* distributions to hysteresis samples of the MFL strains, two sub-populations can be identified for many of the intermediate induction samples. Such a description of the data enables calculation of informative population statistics that can accurately summarise the data for visualisation and modelling.

A major application of flow cytometry is in the identification of sub-populations of cells from biological samples, and in recent years, much effort has gone into automating this process [Bashashati and Brinkman, 2009; Robinson et al., 2012; Ho et al., 2012; Le Meur, 2013]. Such samples can potentially contain dozens of different cell phenotypes and are

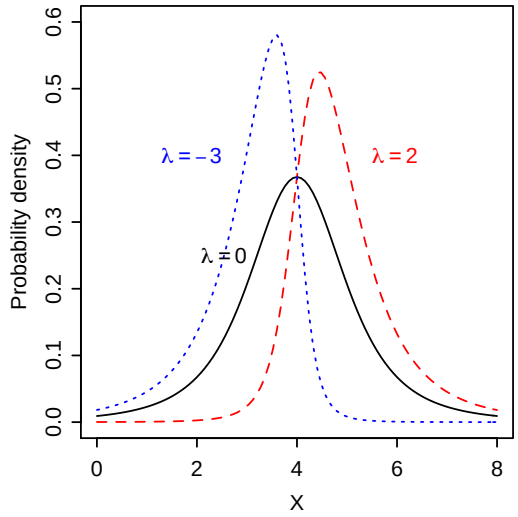
typically probed using a large cohort of fluorophores; the challenge in such instances is in correctly classifying and interpreting a large, high-dimensional data set [Qiu et al., 2011]. The prevalence of such methods provides a strong foundation for analysing the far simpler Tum−CI MFL data. However, the contrasting challenge for the Tum−CI data set is in its low dimensionality: the data suffers from poor resolution. Whereas standard algorithms for studying sub-populations, like *k*-means clustering, work well for samples with well separated sub-populations, here a little more care is required.

The issue of resolving overlapping sub-populations in flow cytometry data was considered some time ago by Lampariello and Aiello [1998]. By characterising the form of a negative control population, Lampariello and Aiello were able to improve the identifiability of a positive control population in spite of significant overlap. Here, a similar approach is followed but with a few modifications and additions that make the analysis more suitable in a quantitative context. These include updating the choice of distribution for describing the data, treating the problem within the more general mixture modelling framework, and generalising the fixed negative control to a family of control distributions whose shape is parameterised as a function of median fluorescence.

Even after Logicle transformation and normalisation, data obtained by flow cytometry tends to have an asymmetric and heavy-tailed distribution that is not well fit by the standard Gaussian (normal) distribution. Whilst the transformation of the data could be further optimised to maximise 'normality' [Finak et al., 2010], an indiscriminate transformation of the data would be unable to account for sample-dependent asymmetries. For example, in *E. coli* the noise in protein expression is typically Gamma distributed [Friedman et al., 2006; Taniguchi et al., 2010] and a Logicle or power transformation of such distributions are not in general well fit by a Gaussian. A more flexible approach is to model the data using a distribution that can account for asymmetry. The Gamma distribution is unlikely to have sufficient flexibility to model all of the Tum−CI MFL data, since instrumental noise and circuit ultrasensitivity would both distort this ideal distribution. The Johnson S_U family of distributions, as chosen by Lampariello and Aiello [1998], includes parameters to introduce asymmetry and fits flow cytometry data well, but its moments (mean, variance, skew, etc.) have a complicated dependence on the parameters. More recently, the skew-*t* distribution has also been identified as particularly suitable for modelling flow cytometry data [Pyne et al., 2009], and is of particular interest here since it introduces skew into the heavy-tailed *t* distribution in a natural way, giving rise to well behaved statistical properties [Azzalini and Capitanio, 2003]. Like the *t* distribution, the skew-*t* distribution has parameters for location (akin to the mean), scale (akin to the variance) and the degrees of freedom, but it additionally includes a shape parameter that can be used to adjust the skew in a positive or negative direction. More information on the skew-*t* distribution can be found in Box 5.1.

An important test of the suitability of the skew-*t* distribution for modelling the Tum−CI

Box 5.1: The skew-*t* distribution. Instructive examples of the skew-*t* distribution shown in Panel A illustrate how the skewness parameter causes the skew-*t* distribution to deviate from the heavy-tailed t-distribution (the case where $\lambda = 0$). A definition of the skew-*t* distribution is provided in Panel B.



A Examples of the skew-*t* distribution for different shape parameters (λ). The other parameters are not varied, setting the location parameter $\mu = 4$, scale parameter $\sigma^2 = 1$ and degrees of freedom $\nu = 3$.

A random variable X follows the skew-*t* distribution if

$$X = \mu + \sigma \frac{Y}{\sqrt{\tau}}$$

where Y is a random variable from the standard skew normal distribution with skewness parameter λ and τ is a random variable from the gamma distribution with shape parameter $\nu/2$ and rate parameter $\nu/2$.
The standard skew normal distribution has a probability distribution function given by $f(y) = 2\phi(y)\Phi(\lambda y)$, where $\phi(y)$ and $\Phi(y)$ are respectively the probability density function and cumulative distribution function of the standard normal distribution.

B Defining the skew-*t* distribution with location parameter μ, scale parameter σ^2, skewness parameter λ and degrees of freedom ν. This (non-rigorous) formulation is adapted from the rigorous definition in Lin et al. [2007].

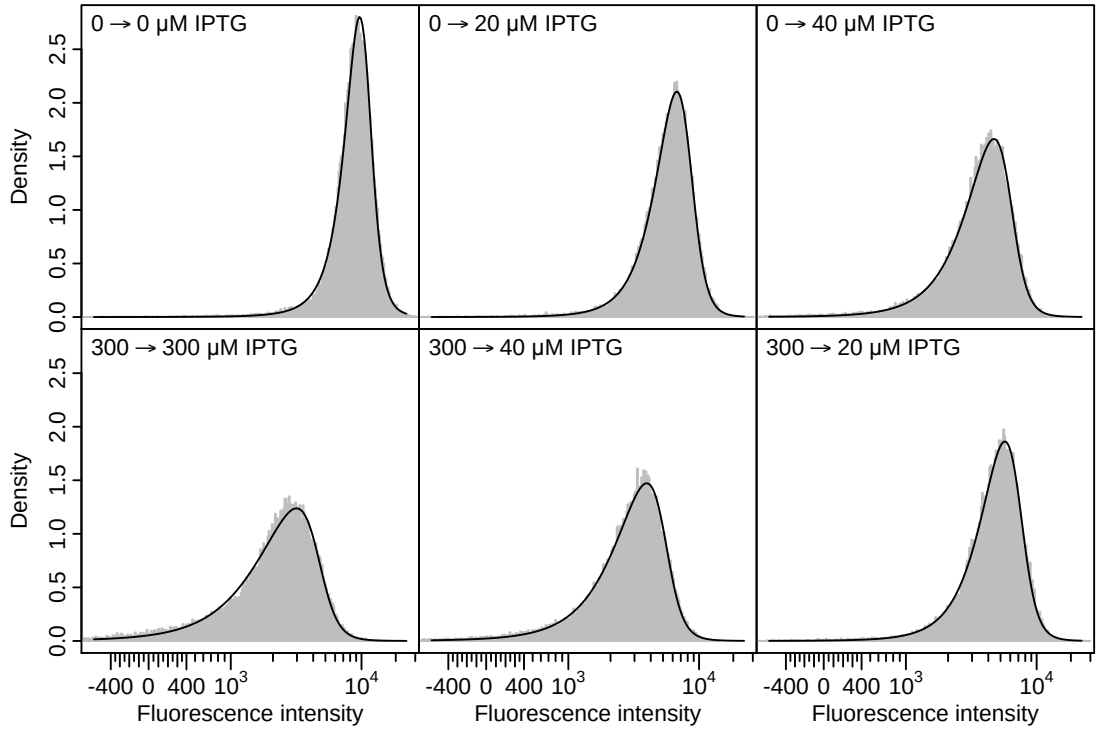


Figure 5.5: The skew-*t* distribution provides a good fit of population fluorescence for the MFL control strains. Shown are histograms of fluorescence intensity measured for ER-MC Tum$^-$ cells grown according to the normal hysteresis assay for a range of initial and final induction conditions as indicated by concentration of IPTG. Fluorescence intensities were normalised according to cell morphology (Section 5.1.4) and are visualised using a Logicle display. The solid black lines are skew-*t* curves that were fit to the Logicle transformed data by Expectation-Maximisation, using the routine provided in the `mixsmsn` R package [Prates et al., 2013].

MFL fluorescence data is how well it fits the MFL control data. Since the control strains lack Tum, they are monostable and hence serve as a useful prototype of the unimodal distribution of fluorescence produced from the *pR* promoter. To this end, skew-*t* distributions were fit to all the control populations measured by flow cytometry using an expectation-maximisation routine from the `mixsmsn` R package [Prates et al., 2013]. The skew-*t* distribution fit all the data remarkably well, and a representative selection of these fits are shown in Figure 5.5 for a range of induction levels. The examples include both minimum and maximum possible expression levels from *pR*, but also intermediate examples starting from both low and high O/N cultures, and together demonstrate the utility of the distribution over the full spectrum of potential *pR* activities. To emphasise the need for a good model of the unimodal populations, it is worth highlighting again that mixtures of the pictured distributions would show significant overlap. In particular, even a mixture of the minimum (300 → 300 μM IPTG) and maximum (0 → 0 μM IPTG) intensity distributions would overlap.

The skew-*t* distribution provides a good fit of the unimodal data, but without additional constraints, it unfortunately still proves to be too flexible for reliably resolving the

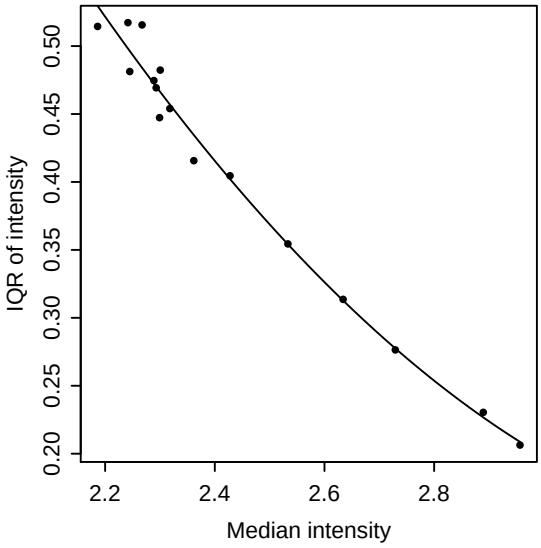


Figure 5.6: The interquartile range (IQR) of intensity for the MFL control distributions varies as a function of the median intensity. Shown plotted against each other are the IQR and median of the normalised and transformed fluorescence for cell samples taken from the normal equilibration hysteresis assay of the ER-MC Tum$^-$ control strain. The solid black line was determined by quadratic regression.

overlapping peaks in the bimodal data sets. An unconstrained fit of the bimodal data can be made using a mixture model of skew-*t* distributions. In statistics, mixture models are used to describe data that has been sampled from a mixture of component distributions, but for which the identity of the component distribution for each observation is unknown. For the Tum−CI MFL, a two-component mixture model of skew-*t* distributions is thus a natural choice. Whilst methods for modelling mixtures of Gaussian distributions have been known for some time, methods for modelling mixtures of skew-*t* distributions have only been developed more recently [Lin et al., 2007]. Mixture modelling is also implemented using the `mixsmsn` R package [Basso et al., 2010; Prates et al., 2013], which treats the skew-*t* distribution as a member of the family of scale mixtures of skew normal distributions, fitting the data using an expectation-maximisation algorithm as in the unimodal case. When applied to the fluorescence data of the Tum−CI MFL, this method of fitting worked well in obvious cases, such as the intermediate case depicted back in Figure 5.2. However, in less obvious cases, to be seen later, it was difficult to determine whether fits represented true bimodality, or if the data was non-identifiable and had merely been overfitted. In some cases, overfitting was clear: samples at the extremes of the hysteresis curve and even control curves often had an improved fit using a two-component mixture. In other cases, non-identifiability was clear: some samples produced markedly different results depending on the initial parameter values chosen. In both cases, the component distributions tended to assume shapes deviating from those seen in the unimodal controls and in cases with obvious bimodality, thus prompting a closer look at the form of the component distributions.

By inspection of the MFL control distributions in Figure 5.5, it can be seen that (in this scale) the spread of intensities tends to be wider for samples with overall low intensity and narrower for samples with overall high intensity. To robustly assess this trend, the

interquartile range (IQR)[5] and median fluorescence of Tum$^-$ control samples were plotted against each other. The result for the normal equilibration time hysteresis assay of the ER-MC Tum$^-$ strain is shown in Figure 5.6. Though nonlinear, the trend is quite striking, particularly for samples with higher intensities; the more abundant low intensity samples are likely to be adversely affected by background instrumental noise. Controls from all other strains and assays showed similarly predictable trends, suggesting that the shapes of the fluorescence distributions might be well-defined functions of promoter activity.

Using median fluorescence as a robust indicator of relative promoter activity, the parameters from skew-*t* fits of the MFL control distributions were indeed found to vary in a predictable way. As exemplified in Figure 5.7, the location parameter showed a linear dependence on median fluorescence, whilst the scale and shape parameters showed quadratic dependences on the IQR of fluorescence, which, as seen previously in Figure 5.6, is in turn quadratically dependent on median fluorescence. The degrees of freedom[6] parameter (ν) was much more variable: it did not show any obvious trends in terms of either median fluorescence or the IQR of fluorescence. When plotted versus the estimated value of the scale parameter, weak associations were observed and the gross trend approximated by a quadratic. This poor result for ν was not entirely unexpected, since ν primarily affects how heavy-tailed the skew-*t* distributions are, and is thus mostly constrained by data in the tails — a considerably smaller fraction of the data set. However, the converse to this poor result is that ν has only a minor impact on distribution shape. In spite of the weak trend observed for the *nu* parameter, skew-*t* distributions that are defined according to parameters estimated from the regression curves in Figures 5.6 and 5.7 describe the data essentially as well as the original distributions with freely-optimised parameters.

So by using the controls as a training set, a family of assay-specific skew-*t* distributions could be defined in terms of a single parameter, the median fluorescence. With such a constraint, the two-component skew-*t* mixture model would be greatly simplified. Instead of 9 parameters, only three would be required: the relative abundance of each sub-population and two location parameters (i.e., the effective median fluorescence values of each sub-population). This amounts to the re-parameterisation of a two-component skew-*t* mixture model according to five assay-specific regression models (based on those shown in Figures 5.6 and 5.7) as derived from unimodal skew-*t* fits of the Tum$^-$ control samples. Such two-component constrained skew-*t* mixture models (and also an analagous one-component model) were implemented using custom R code that could automate the process of deriving parameter regression models from control data, and then calculate the requisite skew-*t* parameters from these regression models. These mixture models were de-

[5]The IQR is the difference between the values of the first and third quartiles (i.e., the 25% and 75% percentiles) of a sample. Unlike the mean and SD, which are sensitive to outliers, the median and IQR are robust measures of location and scale, respectively.

[6]Fractional degrees of freedom (ν) are atypical, but still defined for the skew-*t* distribution. In this context, where the distribution is simply used as a suitably flexible model for data with unknown degrees of freedom, it is left to freely vary.

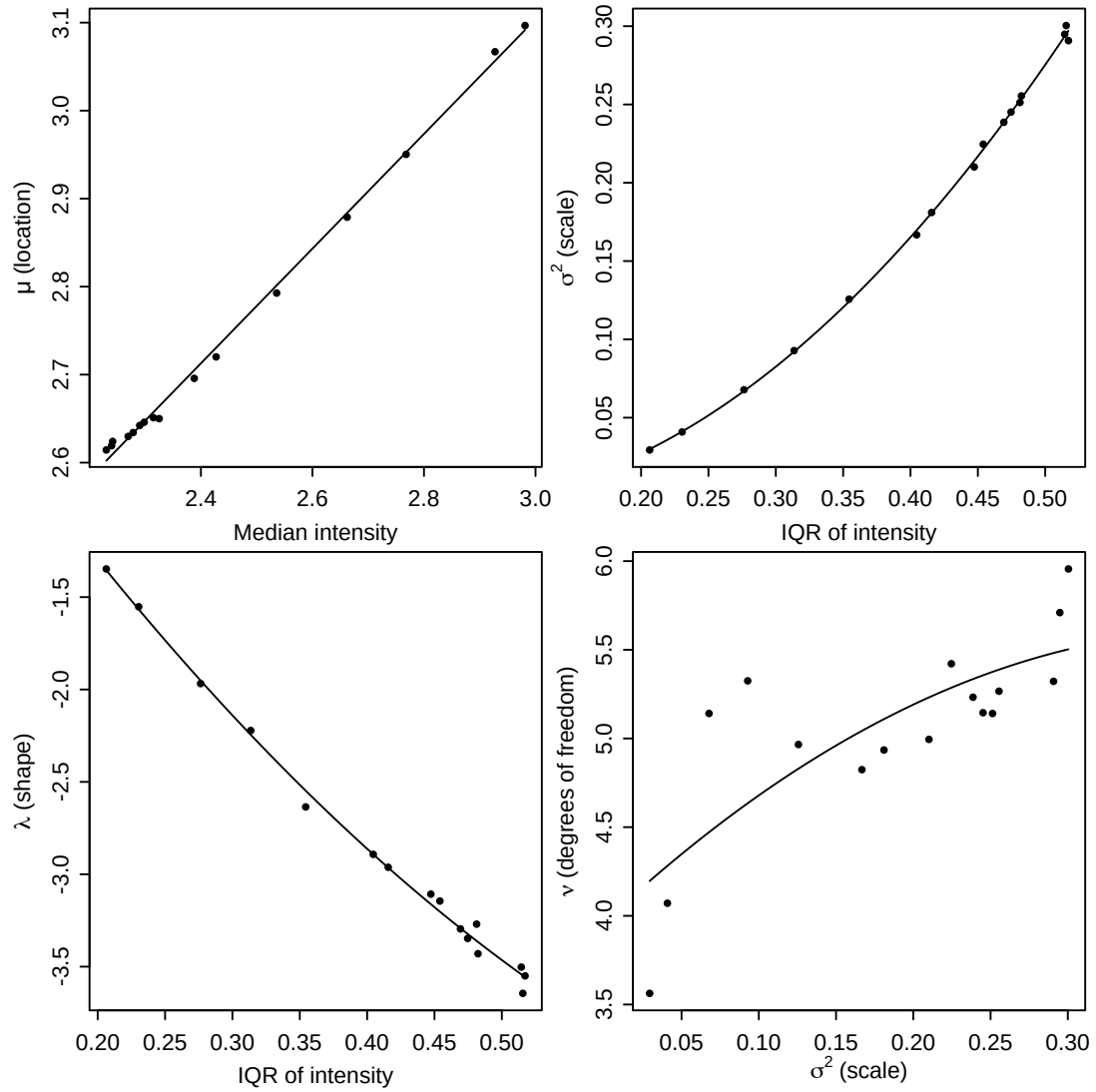


Figure 5.7: The fitted skew-*t* parameters of the MFL controls vary as functions of the median and the interquartile range (IQR) of the intensity. The normalised and transformed fluorescence intensities of cell samples taken from hysteresis assays of the ER-MC Tum$^-$ control strain were fit with skew-*t* distributions. From the resulting set of fits, the four parameters defining the skew-*t* distribution (location (μ), scale (σ^2), shape/skew (λ) and degrees of freedom (ν)) are shown plotted versus either the median or IQR of the samples' fluorescence intensities. In the case of the degrees of freedom parameter (ν), a somewhat more visible trend was observed with the predicted value of the scale parameter (σ^2), so that is shown instead. Regression lines are shown for each plot: the location parameter was fit using linear regression; all other parameters were fit with an additional quadratic term.

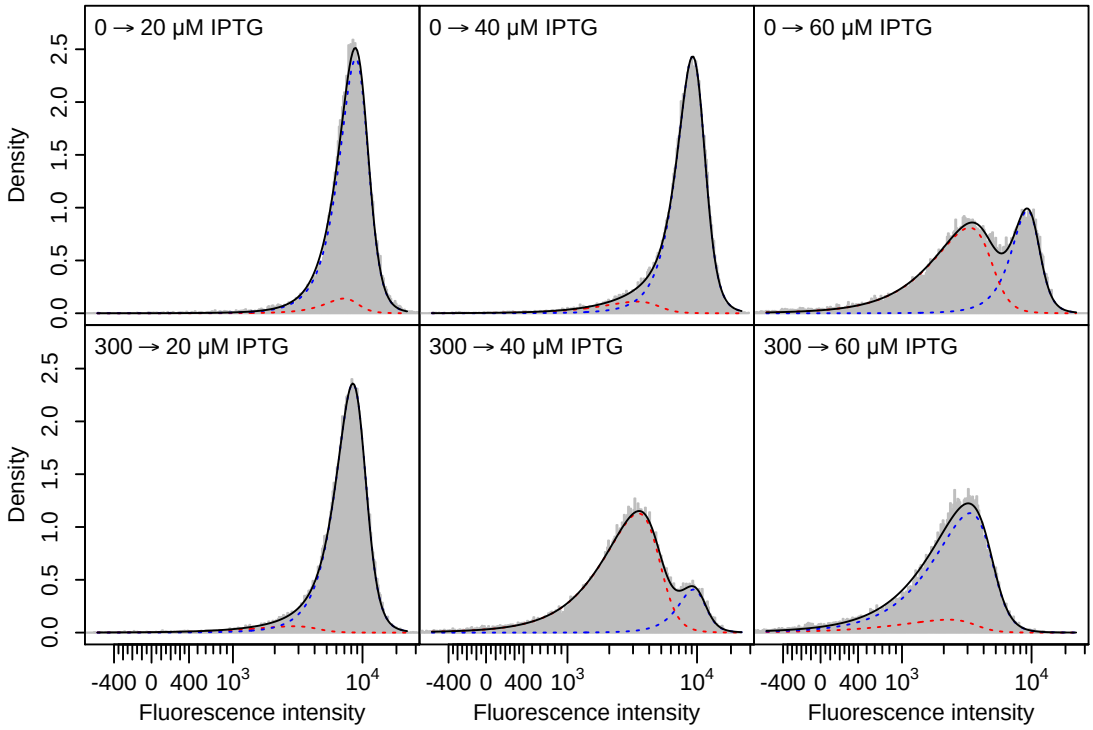


Figure 5.8: MFL distributions in the bistable region are a bimodal mixture of control-like subpopulations. Shown using a Logicle display are histograms of the normalised fluorescence obtained for samples with a range of initial conditions and IPTG induction levels (as indicated) from the normal equilibration time hysteresis assay of the ER-MC Tum$^+$ MFL strain. The fluorescence distributions were fit by two-component mixture models made up of constrained skew-*t* distributions; constraints were imposed using a set of regression models, as described in the text, to restrict the skew-*t* distribution to shapes matching the same assay's Tum$^-$ control distributions. The density for each fitted mixture model is shown overlaid on its respective histogram as a solid black line; the dotted red and blue lines indicate the low- and high-intensity component distributions respectively.

fined according to the controls for each flow cytometry hysteresis assay, and then one- and two-component models were fit to all the Tum$^+$ MFL sample data using maximum likelihood estimation. To further control the parameter space, the median for the high intensity component distribution was written as a fold ratio of the lower. Constrained maximisation of the likelihood could then be employed to restrict this ratio to a value greater than one and also restrict the relative proportion of the components to a value between 0 and 1. In spite of the reduced number of parameters and the additional constraints, this new mixture model still matched all of the Tum−CI MFL fluorescence distributions surprisingly well.

The constrained skew-*t* mixture model offers sufficient flexibility to accurately describe all of the Tum−CI MFL distributions. Examples of two-component mixture model fits of fluorescence data obtained for normal equilibration time hysteresis assays of the ER-MC MFL strain are shown in Figure 5.8. For some samples, bimodality is clear (e.g., 0 → 60 μM IPTG and 300 → 40 μM IPTG) and the locations of the two component distributions appear well-defined. Conversely, for samples where a unimodal distribution would seem

more appropriate (e.g., 0 $\rightarrow$ 20 μM IPTG and 300 $\rightarrow$ 60 μM IPTG), the two-component model is only able to fit the data using essentially overlapping component distributions. In these cases, the component distributions have reduced identifiability, indicating that the one-component model would be preferred. These extreme cases illustrate how the constrained mixture model might be used to robustly assess the number and location of component sub-populations.

Fringe cases, however, are better handled using a formal analysis of goodness of fit. Such cases are exemplified in Figure 5.8 by the 0 $\rightarrow$ 40 μM IPTG and 300 $\rightarrow$ 20 μM IPTG samples. It is instructive to start by considering how the two-component and one-component models compare. Since adjusting the median location is the only way to adjust peak shape in the constrained skew-*t* distributions, the optimal fit for a one-component model will be very similar to whichever component is dominant. In the stated examples, this dominant component distribution is the high intensity one (dotted blue lines in the figure), and this component accounts for most of the fluorescence. However, the less abundant distribution clearly makes a small but significant difference in each case. A more rigorous evaluation of peak fit can be made using a log-likelihood ratio test between the one- and two-component models. For this test, the null model is the one-component model, since it is a subset of the two-component model. A useful result on the corresponding test statistic (the log-likelihood ratio) is that its null distribution tends to a χ^2 distribution in the limit of large sample sizes (see Rice [2007, sect. 9.4]). By assuming this result for the mixture models here, *p*-values were calculated to compare the significance of one- and two-component models for the maximum likelihood estimates of the Tum^+ MFL fluorescence distributions.

The constrained skew-*t* mixture model provides high-confidence confirmation for the presence of two sub-populations in the bistable region of the MFL strains. Most of the known and presumed unimodal populations (i.e., controls and ends of the hysteresis loop) presented with *p*-values greater than 0.001 (in fact, with *p*-values often indistinguishable from 1), indicating non-significance for the two-component model. In contrast, populations displaying bimodality, including the fringe cases, all presented with very low *p*-values (less than 1×10^{-16}), indicating high significance for the two-component model in these cases. However, direct use of the *p*-value as a threshold would result in more type II errors than desired (i.e., where the one-component model would be rejected when it should not be), as judged by the test results for the (unimodal) control distributions. Much better discrimination was achieved by setting a more conservative threshold directly on the computed log-likelihood ratio[7]. This threshold was set to a value that would avoid

[7] This approach to assessing goodness of fit is very similar to the use of the Akaike Information Criterion (AIC) or Bayesian Information Criterion (BIC), except that no penalty is applied for overfitting as the number of parameters is increased. Here the penalties of the AIC or BIC are not applied, since (1) the number of parameters is very small in comparison with the size of the data set, so the corrections are minor (particularly for the AIC), and (2) the distributions under consideration (skew-*t*) are not from the exponential family (breaking an assumption of the BIC).

any type II errors for the control distributions; taking the 0.0001 quantile for a Gaussian fit of the control ratios was generally sufficient. The need for such a conservative threshold was due to both the large number of events sampled for each experimental condition (tens of thousands), and a handfull of distributions (mostly at the extremes of the fluorescence intensity spectrum) that were not fit by the constrained skew-*t* distribution quite as well. However, since these anomalous outliers were avoided by using a stricter threshold, the constrained skew-*t* mixture model was nonetheless a high-confidence predictor for bimodality.

As a brief aside, the morphology normalisation procedure introduced in Section 5.1.4 was essential to the robust behaviour of the constrained skew-*t* mixture models. Without normalisation, the trends of fitted skew-*t* parameters with median fluorescence and IQR (i.e., from the equivalent plots to those in Figure 5.7) became much more variable. This greatly decreased the efficacy of the subsequent two-component model, which could no longer replicate the close match of this model to the bimodal distributions that was seen for the normalised data. Accuracy and confidence in the fits were much reduced, emphasising the importance of the normalisation for rendering the data amenable to predictable statistical analysis in terms of the well-defined skew-*t* distribution.

Given that location is the only parameter required to specify the constrained skew-*t* distribution, fits of bimodal data sets using the two-component model produce a highly sensitive quantitative measure of the location of each sub-population. Since these constrained skew-*t* densities are also characteristic of a monostable distribution, the close match of such models to the data strongly indicates that the two sub-populations are distinct. That is, each sub-population arises in a separate zone of stability of the gene network, and only a small number of cells are 'in transition' between these two zones. How might such a mixture of populations arise? The deterministic model would predict a unimodal population of cells with a history-dependent location, so clearly there must also be some level of stochasticity that enables transitions between the two stable zones. That is, the synthetic Tum−CI MFL is not strictly bistable, but is instead comprised of two semi-stable sub-populations.

5.3 The mixed population model reveals the stable states predicted for the MFL strains

Assays of the Tum−CI MFL strains by flow cytometry have now revealed that many of the hysteresis cultures were in fact made up of a mixture of two sub-populations. Systematic variation in the presence and size of these sub-populations as a function of CI production rate was already evident in the curated flow cytometry data presented in Section 5.1.5, but without quantitative tools for describing the sub-populations, it was difficult to draw concrete conclusions. This prompted the development of the constrained skew-*t* mixture

model in Section 5.2, which was fitted to the data to enable both robust identification of sample modality (i.e., unimodal or bimodal), and precise, high-confidence estimates of the locations and relative proportions of each sub-population. In this section, the quantitative insights made possible through such a detailed statistical model are exploited to search for identifiable trends in the sub-populations across the hysteresis loop. With a view towards reconciling the data with the already established deterministic model of hysteresis, the section begins by briefly considering the conditions under which it is suitable to make a deterministic approximatation to a stochastic process. This lays the groundwork for interpreting the constrained skew-*t* mixture model as the projection of stochastic data onto deterministic components, thus motivating a natural experimental state descriptor: the expected fluorescence intensity of each component distribution. When applied to the long equilibration time hysteresis assays of each of the Tum−CI MFL strains, this deterministic projection helps to paint a much more informative picture of stability in the hysteresis assay than the whole-population average results could. The mixed population long equilibration time hysteresis curves expose the stable states in the Tum−CI MFL strains, making it possible to interpret their behaviour in terms of the steady-state MFL model developed in Section 2.3.

Deterministic models approximate a full stochastic process by considering only the time evolution of the expected or 'average' values of that process [Kaern et al., 2005; Wilkinson, 2006]. Such an approximation forms the basis for comparing deterministic models directly with LacZ assay measurements, which are implicitly whole-population average activities. In flow cytometry, however, promoter activities are measured for each cell in a culture, and each measurement can be considered (roughly speaking) to be an independent realisation of a 'noisy' or stochastic process. Stochastic processes cannot in general be described by a purely deterministic model, but when considered near macroscopic (i.e., deterministic) stable states, valid approximations for the shape of the stochastic distribution can be made by series expansion around the deterministic solution [Thattai and van Oudenaarden, 2001; Elf and Ehrenberg, 2003; van Kampen, 2007]. In this way, a deterministic model can remain useful for *local* analyses of behaviour and noise, provided the system under consideration has well-behaved[8] steady-state solutions and is close to equilibrium.

The Tum$^-$ control strains have a single steady-state solution and are well suited to deterministic modelling. Indeed, deterministic models of CI−*pR* repression have proven effective in the past [Shearwin and Egan, 1996; Dodd et al., 2007b] and even the behaviour outside steady state is relatively well explained by the deterministic model (see Section 4.3). As such, the fluorescence distributions of the Tum$^-$ control strains could be considered a prototype for the stochastic spread of an otherwise 'deterministic' population. By charac-

[8] A well-behaved steady-state solution means that it must be stable in all dimensions, excluding saddle point solutions for example.

terising the MFL samples in terms of mixtures of this prototype distribution, samples are effectively split into putative deterministic components. Since two stable states are predicted for the Tum−CI MFL according to the steady-state deterministic model, it stands to reason that this gene network is suited to such a localised analysis. The goodness of fit of the bimodal mixture models to the Tum−CI MFL samples further validates a local deterministic interpretation of each sub-population, even if stochastic mechanisms must describe the presence of a second population.

For the long equilibration time MFL samples, each component of the fitted mixture model of constrained skew-*t* distributions can be treated as a local estimate of the noise distribution near each deterministic steady state. The expected value (weighted average) of each component distribution then provides a good approximation for its respective deterministic steady-state solution. This expected fluorescence intensity can be calculated directly for each sub-population from the fitted constrained skew-*t* distributions. The constrained skew-*t* distributions are specified in terms of a single parameter representing the expected median fluorescence for each sub-population. The median fluorescence is a useful measure of location for sampled data since it is robust to outliers, but for the fitted distributions, the expected fluorescence is calculated since it is more in keeping with a deterministic approximation of the stochastic data set. Since cellular fluorescence is measured on an untransformed scale, whilst the distributions are fitted on a Logicle scale, the expected fluorescence should be calculated as the expected value of the Logicle function over the skew-*t* distribution of interest. The Logicle function ($\mathcal{L}\colon X \to I$) gives the fluorescence intensity (I) as a function of the transformed data[9] (X). So if the sub-population is described by a skew-*t* distribution with probability density function $\rho(x)$, then the expected fluorescence intensity is given by:

$$E[I] = \int_{-\infty}^{\infty} \mathcal{L}(x)\rho(x)\mathrm{d}x \tag{5.1}$$

The skew-*t* probability density, $\rho(x)$, is population-specific and specified using parameters drawn from the control curve regression models (see Section 5.2). The Logicle function is assay-specific, and defined according to an optimised width parameter (see Section 5.1.2). The mathematical expressions for the skew-*t* probability density and Logicle function make analytical integration of Equation (5.1) nontrivial, so numerical integration (implemented in R; see Appendix B) is used for calculating the expected fluorescence intensity. The resulting expected fluorescence value is a natural deterministic representation of each sub-population, comparable with the *pR* activities that might be reported by LacZ assay.

[9] This declaration of the Logicle function (i.e., a function giving the dependence of *data* units on *display* units) is as defined by Parks et al. [2006]. Note, however, that it is the *inverse* of the Logicle function that is used to perform a 'Logicle transform'. This misnomer is in common usage since the inverse of the Logicle function cannot be expressed analytically.

A given hysteresis sample can now be completely specified from its fitted mixture model as an optimal number of mixing components (i.e., one or two), the fractional occupancy of each sub-population relative to the whole, and the expected fluorescence (*pR* activity) of each component. As described in Section 5.2, the log-likelihood ratio is used to decide whether a sample is unimodal or bimodal. For unimodal samples, there is only one component with an occupancy fraction of 1; in this case, the expected fluorescence of the underlying skew-*t* distribution represents that of the whole population. For bimodal samples, there are two components, each present as a fitted fraction of the whole population, and each with a distinct location given by the expected fluorescence intensity of the respective skew-*t* distribution.

These state descriptors are shown plotted against the hysteretic variable (CI production rate) in Figure 5.9 for samples coming from long equilibration time hysteresis assays of each of the candidate Tum−CI MFL strains (i.e., the WR-LC, WR-MC, ER-LC and ER-MC strains). Dotted lines are used to approximately indicate the CI production rates over which samples are expected to transition from a bimodal to a unimodal structure. Because of day-to-day variation in flow cytometer performance, the units of fluorescence are consistent within each assay, but are not comparable across different assays. Furthermore, the fluorescence measurements introduce a nonzero offset resulting from cellular autofluorescence and, to a lesser degree, from the background noise of the optics. However, as was shown previously in Figure 5.3, the fluorescence units match the LacZ assay units to within an offset and scaling factor.

The high intensity populations appear to show greater variability in intensity, which is presumably compounded by the way in which the constrained skew-*t* distributions are constructed and transformed. Since the constrained skew-*t* distributions are defined in terms of the controls, the lower abundance of high intensity control samples means that the regression models provide lower confidence estimates of the skew-*t* parameters at those intensities. Furthermore, since the distributions are fit to data in the Logicle scale, any errors in the fitted population locations will be amplified by the transformation back to the original scale. Confidence in the expected intensity of a sub-population is also lower for those occupying a smaller fraction of the sample. In spite of all these potential sources of error, the Tum^- control data are remarkably well-formed for a single replicate, and the MFL data display convincing trends on the whole.

Each of the MFL strains exhibits a region of CI production rates in which bimodal cell populations can be found. As can be seen in Figure 5.9, these bimodal regions are (for the most part) flanked to either side by unimodal samples. For the bimodal samples, the separation in activity between low and high fluorescence sub-populations is comparable to the separation in activity between the flanking unimodal populations. This is a striking contrast with the separation seen previously between the low and high O/N long equilibration time LacZ assay curves in Figure 4.3 of the previous chapter. A cursory inspection

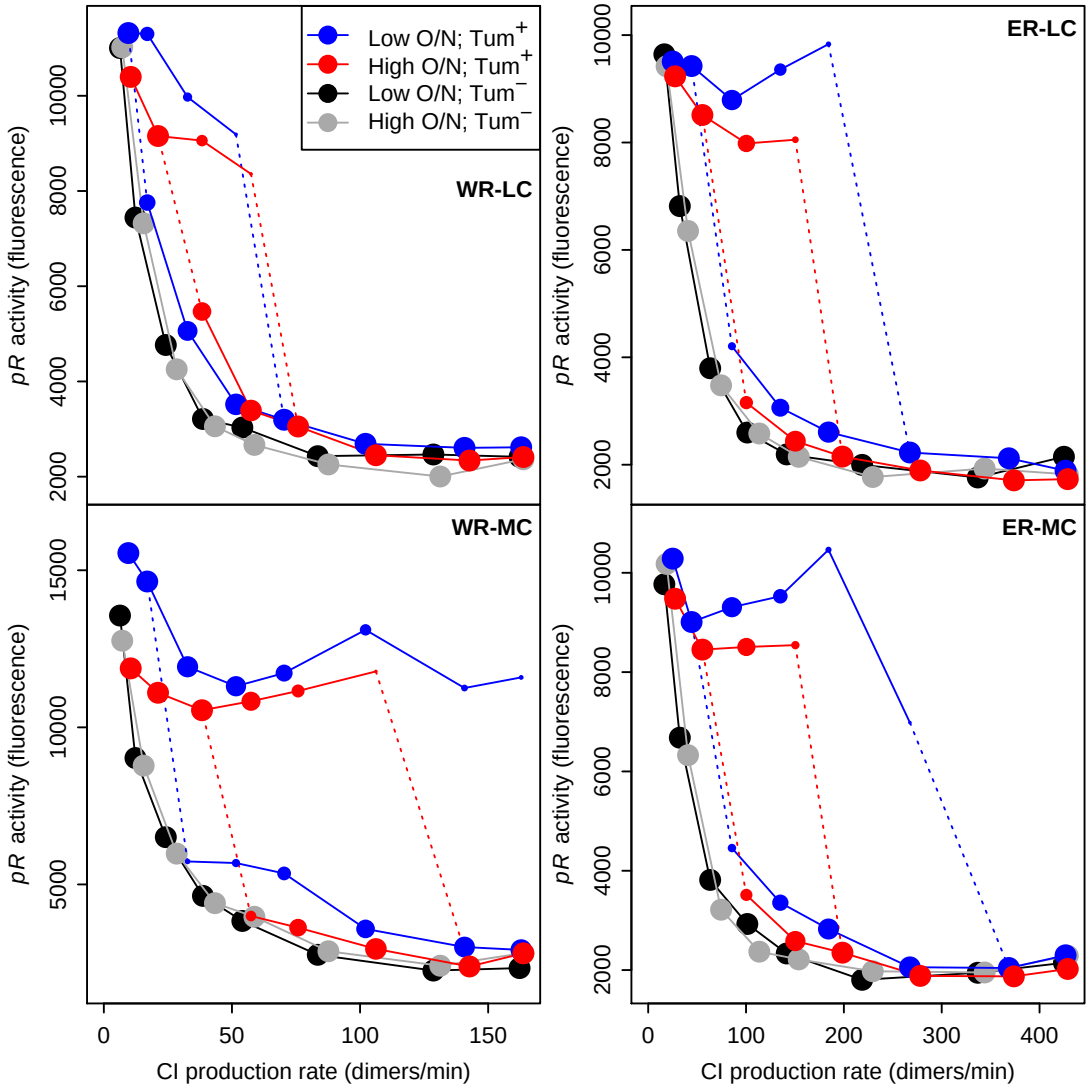
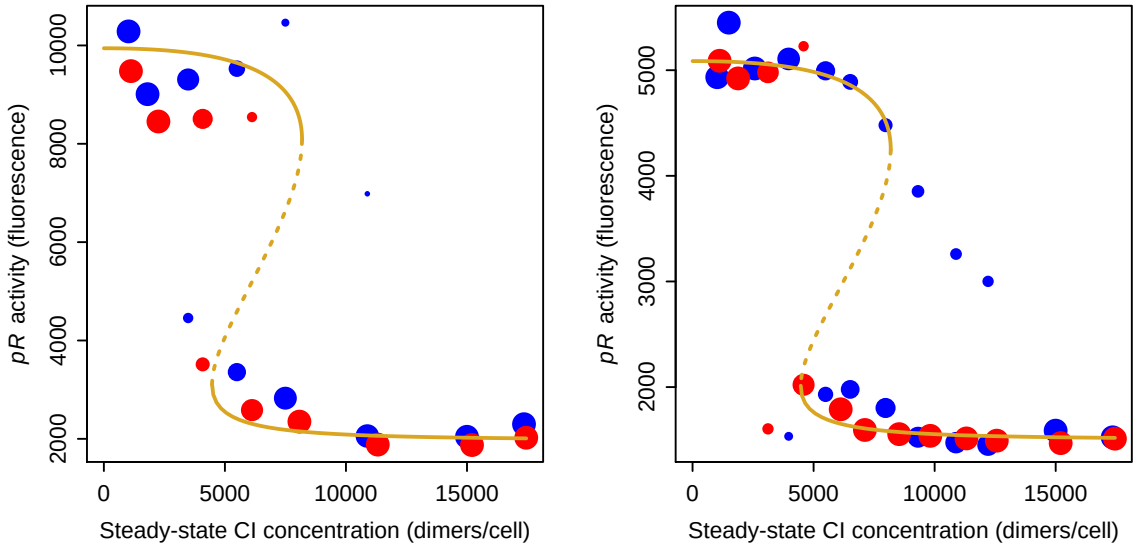


Figure 5.9: Bimodal cell populations are found in each of the Tum−CI MFL strains. Long equilibration time hysteresis assays were performed for each of the MFL strains (WR-LC, WR-MC, ER-LC and ER-MC) and the *pR* activity of these samples measured by flow cytometry for 10,000–30,000 cells. These population fluorescence measurements were curated as described in Chapter 5 and fit with mixture models of constrained skew-*t* distributions. Shown plotted are the expected values of the component distributions versus the final CI production rates (normalised from the final IPTGs as per Section 3.3). Values for samples originating from high or low overnights (O/Ns), and experimental (Tum^+) or control (Tum^-) strains can be distinguished by colour according to the legend. Samples were categorised as unimodal or bimodal using a threshold on the log-likelihood ratio between the alternative models; for each CI production rate, unimodal samples are plotted as a single expected fluorescence value, whilst bimodal samples are plotted with two. Dotted lines mark the transitions between unimodal and bimodal descriptions of the samples. The area of each point is proportional to the fraction of that sample that belongs to the respective component distribution. Note: for reasons described in the text, the identity of the ER-LC strain used for this assay came into question so the results shown here for that strain should be treated accordingly.



(a) Low quality **(b)** High quality

Figure 5.10: The results obtained by flow cytometry can be interpreted in terms of the steady-state deterministic model. The low and high O/N Tum$^+$ sub-population location data from long equilibration time ER-MC hysteresis assays are reproduced here overlaid with a steady-state curve derived from the Tum−CI MFL model. The model curve is essentially as previously depicted in Figure 2.7(b), but the model parameters were manually adjusted to improve consistency with the data ($\varepsilon_R = 1.2 \times 10^3$ dimers/cell; $\mathrm{H}_R = 2.0$; $\varepsilon_S = 1.1 \times 10^4$ monomers/cell; $\mathrm{H}_S = 1.8$; $\mathrm{M}_T = 7.5 \times 10^4$ monomers/cell). Due to day-to-day variation and nonzero autofluorescence, the model curves have assay-specific maximum fluorescence parameters and assay-specific offset parameters. **(a)** Data obtained from ER-MC panel in Figure 5.9 ($\mathrm{M}_Z = 3.6 \times 10^3$; Offset = 1.5×10^3). **(b)** Higher quality data to be presented later in Figure 5.11 ($\mathrm{M}_Z = 8.0 \times 10^3$; Offset = 2.0×10^3).

of the shapes outlined by the sub-populations also suggests much more favourable comparison with the hysteresis curves modelled deterministically in Section 2.4.2.

The deterministic model cannot explain how bimodal populations arise in the hysteresis assay, but it can still serve as a useful framework for interpreting the behaviour of each sub-population separately. A steady-state analysis of the Tum−CI MFL network (Section 2.3.1) revealed how it could support two stable states in the bistable region. When out of equilibrium in the region of bistability, whether by an extrinsic change to CI production rate or by the effects of intrinsic noise, the positive feedback loop acts to bring the system towards the 'nearest' stable state. This reveals how the Tum−CI MFL circuit might maintain a bimodal population, since even with the presence of noisy gene expression, the stable states would act as competing basins of attraction that increase the probability of cells occupying states near the stable ones. A comparison of the steady-state model and two independent long equilibration time hysteresis assays of the ER-MC strain are shown in Figure 5.10, demonstrating how the two sub-populations might co-locate with each stable state in the bimodal region. The data shown in Figure 5.10(a) is from Figure 5.9, whilst the data in Figure 5.10(b) is from a higher quality assay that will be introduced later in Section 5.4 and includes measurements at more CI production rates and with higher

cell counts per sample. For the figure, the parameters chosen for the original steady-state model in Section 2.3.1 were manually adjusted to improve the match between model and data. Apart from scaling (via the M_Z parameter) and offset parameters, the model curves are identical for each assay. Though not a fitted model, the good match between the model and data shows that the steady-state structure of the MFL is now better revealed after projection of the stochastic results onto prototypical deterministic components.

When the *rate* of stochastic switching between the two zones of stability is equal, the sizes of the sub-populations are an indication of the relative stability of each stable state [van Kampen, 2007]. For both low and high O/N cultures, the high intensity sub-populations consistently become smaller as they near and extend beyond the right-most bifurcation boundary, whilst the low intensity sub-populations consistently become smaller as they approach the left-most bifurcation boundary. By referring back to the steady-state analogy depicted in Figure 5.10, feasible explanations for these observed decreases in stability near the bifurcation boundaries include (1) a decreased vertical distance to the unstable state (dotted line), making it easier to cross into the alternative basin of attraction, or (2) horizontal proximity to the monostable region, making it easier to cross into the alternative monostable state. Both mechanisms are likely to contribute to the observed behaviour to varying degrees. A stochastic model of the MFL circuit will be introduced in the following chapter to assess how noise in the vertical (Tum production) or horizontal (CI production) directions might affect stability in the Tum−CI MFL.

Differences between the hysteresis assays for each of the strains can also be rationalised in terms of the steady-state model. As previously realised for the LacZ assay results, the difference between the wRBS and eRBS strains is primarily a different scale of CI production rates on the x-axis[10]. Where the wRBS should reveal more detail for lower production rates, the eRBS should extend the hysteresis curve to higher production rates. This can be seen when comparing the WR-MC and ER-MC strains: the WR-MC hysteresis curve is a 'magnified' version of the ER-MC hysteresis curve. Doubts over the identity of the ER-LC strain[11] preclude a similar comparison with the WR-LC strain. The effect of *pR-tum* plasmid copy-number can be seen by comparing the WR-LC and WR-MC assays: the low-copy *pR-tum* plasmid produces a much smaller region of bimodality. This corresponds well with the predictions of the steady-state model where a reduction in the unrepressed level of Tum (M_T) was shown to decrease the size of the bistable region in the model (Figure 2.9(a)). The region of bistability was also predicted to shift to lower CI production rates, and this is consistent with the data. These results further highlight how the projec-

[10] Note that since the CI production rates in the figure have been scaled according to low confidence results obtained by western blot, it is not surprising that the eRBS and wRBS production rates appear inconsistent.

[11] The surprising similarity between the ER-LC and ER-MC results, and the larger bimodal region of the ER-LC strain compared with the WR-LC strain, brought into question its identity in this assay. Further investigation suggested that whether by contamination or experimenter error, it is likely that the strain being assayed was the ER-MC strain. The ER-LC assay will be ignored in the discussion; it is left in the figure for completeness.

tion of stochastic data onto deterministic components can facilitate analysis in terms of a steady-state deterministic model.

The semi-deterministic interpretation of Tum−CI MFL bistability corresponds well with the basic trends of the hysteresis data shown in Figure 5.9, but the treatment of the samples as pure two-component mixture models is limited. The two-component model is clearly a simplification of the full stochastic model, which must further account for some level of cells that are stochastically 'switching' between the sub-populations. Though the bimodal distributions present as remarkably good fits of the data, the close overlap of the peaks could easily obscure an intermediate population of cells. If the size of this intermediate population was comparable with the sizes of the steady-state populations, the mixture model components would be forced to compensate for the intermediate population. In that case, the expected intensity of each population would tend to become more centralised. There is evidence that such compensation may have occured for certain samples in the hysteresis assays. This is particularly obvious for the high fluorescence sub-populations in the WR-MC low O/N samples of Figure 5.9, where the sub-populations in the middle of the bimodal region appear to 'pinch' together. It would be difficult to correct for this centralising bias[12], but it is important to keep it in mind when interpreting the results.

Before moving on to a more involved stochastic treatment of the data, it is worth exploring an additional dimension of the deterministic model that has not yet been considered: the equilibration time. In the following section, the different equilibration time assays will be used to study the dynamic evolution of the Tum−CI MFL. For this more detailed analysis only one strain will be considered. The WR-LC and (presumably) the ER-LC strains both have smaller regions of bimodality, making them harder to study quantitatively. In contrast, the bimodal region of the WR-MC strain is so large that the hysteresis curve is truncated at the highest CI production rate; the fitted components of this strain also showed particularly pronounced pinching. As such, the ER-MC strain stands as the most suitable for further investigation and time course measurements of that strain will be analysed and discussed in the coming section.

5.4 Noisy switching between sub-populations occurs throughout the hysteresis assay

Flow cytometry measurements of the Tum−CI MFL at the long equilibration time point revealed a bimodal population structure for hysteresis samples in the bistable region. Under the equilibrium assumption, the long equilibration time assays could be interpreted

[12] In theory, the bias could be corrected for by adding a third component to the mixture model to account for these intermediate cells. However, its shape would be unlikely to conform to the constrained skew-*t* model, and, being located between two closely overlapping fluorescence distributions, it would also be hard to resolve with satisfactory confidence.

in terms of the steady-state deterministic model to show how the sub-populations might co-locate with the stable states. However, without measurements at other time points, it is not possible to judge how close the MFL samples are to equilibrium, so the quality of the steady-state comparison remains in doubt. In this section, flow cytometry measurements of the short, medium and long equilibration time hysteresis assays of the ER-MC Tum−CI MFL are used to investigate the approach to equilibrium. The mixture model components reveal a much slower rate of change in *pR* activity than seen using the LacZ assay. The sub-population sizes further indicate that active stochastic switching between the stable states occurs throughout the time course, which augments the apparent rate of equilibration. A majority of the state switching in the low O/N samples appears to occur before the short equilibration time, whereas the high O/N samples show substantial switching between the times assayed.

Different equilibration time points of the hysteresis assay were previously measured by LacZ assay by modifying the protocol for log-phase growth. Whilst measurement by flow cytometry removes the need to maintain consistent culture OD_{600}s, the same approach was employed here to maximise comparability with the original whole-population assays. For simplicity, only the ER-MC strain was considered in this more extensive analysis. Using the data analysis methods developed earlier in the chapter, the measured populations are projected onto deterministic components to reveal bimodal populations at all equilibration time points. The data further reveals that switching between the constituent sub-populations occurs throughout the observed time period, particularly for the high O/N samples.

Two independent sets of equilibration time assays were measured by flow cytometry. In the first set, short and normal equilibration time hysteresis assays were performed for the ER-MC strain to complement the long equilibration time assay shown in Figure 5.9. These assays were analysed as described in Section 5.3 and are shown in the upper panels of Figure 5.11. For the second assay set, two minor modifications were made to the protocol to improve data quality. Firstly, to obtain greater detail in the bimodal region of the MFL curves, additional intermediate induction levels were included for the Tum^+ samples at the expense of fewer induction levels for the Tum^- controls. Secondly, to improve the quality of the morphology normalisation, the number of cells recorded per sample was substantially increased and greater care taken to ensure FSC/SSC stability during the course of measurement. These changes meant that the average cell density could be reliably applied in the normalisation protocol (as previously noted in Section 5.1.4; also see Appendix B), which both increased consistency for the expected intensities of the normalised samples and, perhaps more importantly, increased the quality of the regression models used to define the constrained skew-*t* mixture model. The analysed data from the resulting second set of short, medium and long equilibration time assays is shown in the lower panels of Figure 5.11. Whilst the high cell count assay set (lower panels) is of higher

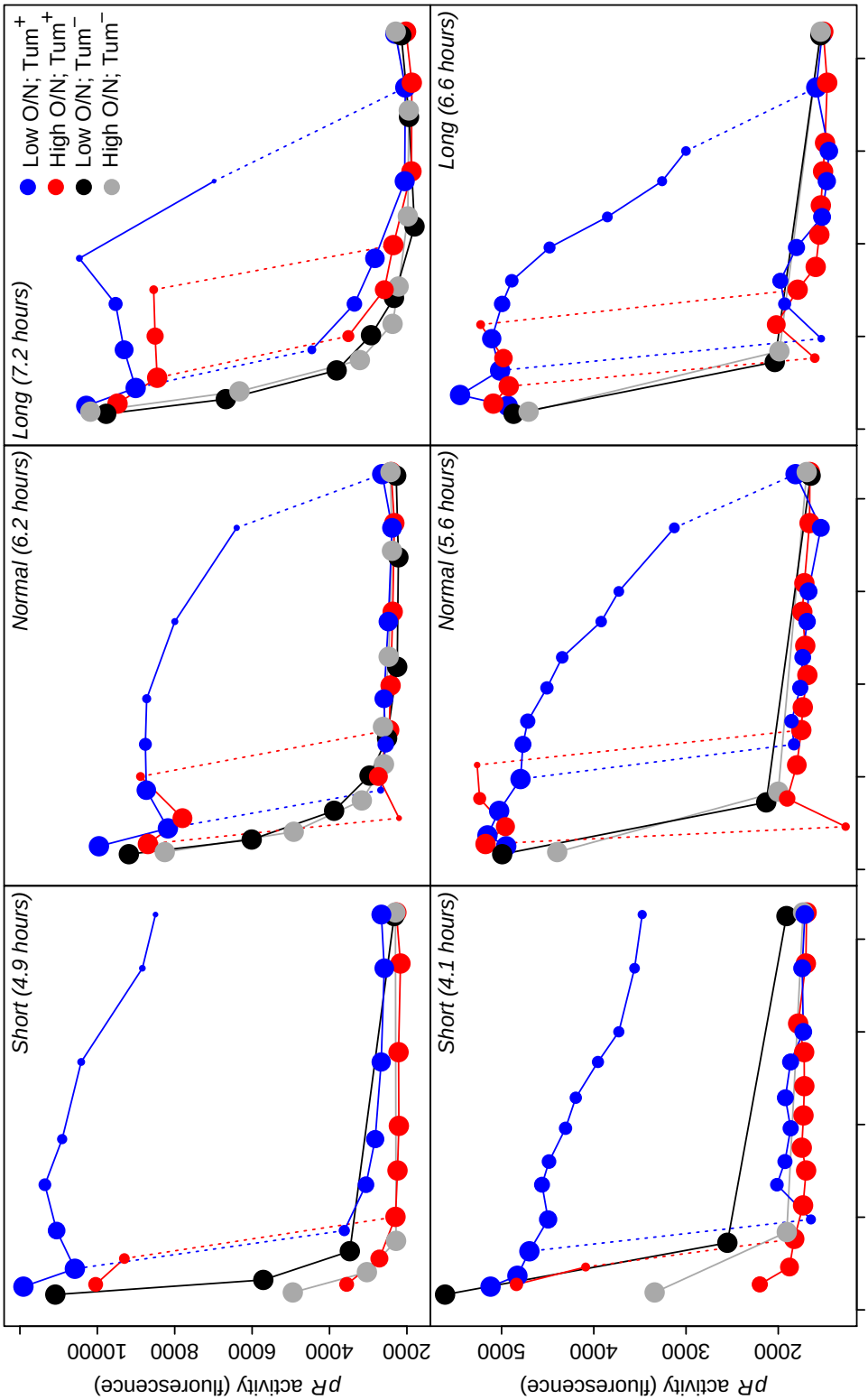


Figure 5.11: Bimodal populations are found at all the equilibration time points tested by hysteresis assay for the ER-MC Tum−CI MFL. *Continued on following page.*

quality, it is still complemented by the low cell count assay set (upper panels), and both sets will be analysed together.

From a deterministic perspective, equilibration of the samples is judged by changes in mean activity over time. Crudely speaking, the mean activities of the sub-populations might be expected to follow something akin to a deterministic trajectory, and, to a limited degree, such equilibration behaviour appears to occur for the ER-MC MFL. Clearly, the unimodal Tum$^-$ controls equilibrate quickly, being already quite close to equilibrium at the normal equilibration time point. The dynamic trajectories of the MFL sub-populations are harder to interpret, but starting from relatively flat lines at the short equilibration time, they do progressively fold towards the presumptive steady-state curve by the long equilibration time. The rate of collapse of these sub-populations is slower than was observed by whole population assay, and it seems likely that some additional equilibration time might help to further tighten the shape. If so, then the example steady-state curve overlaid on the long equilibration time assays back in Figure 5.10 may be slightly too broad. In any case, with stochastic mechanisms clearly contributing to MFL behaviour, it is unlikely that the dynamic deterministic trajectory can be accurately matched to the sub-population means[13] and it becomes equally important to consider the rate of stochastic switching between the zones of stability.

As is evident from Figure 5.11, bimodal populations are found at all equilibration time points measured for the ER-MC MFL. At the short equilibration time point, the low O/N samples are bimodal at all but the lowest CI production rates, indicating that much of the split into two populations must occur before this time. In contrast, the high O/N samples at this time point only show bimodality at the lowest CI production rates. The CI production rates at which bimodality is observed then change gradually at subsequent equilibration times. For the low O/N samples, bimodal populations are lost at the highest CI production rates, with no obvious changes at lower CI production rates. For the high O/N samples, the location of the bimodal populations shifts to higher CI production rates over time. This sets a clear contrast between the alternative O/Ns: whereas most of the low intensity sub-populations of the low O/N samples are already present by the short equilibration time point, most of the high intensity sub-populations of the high O/N samples are only observed after the short equilibration time. The implication is that, in general, the high intensity sub-populations are less stable than the low intensity ones.

With regards to relative sub-population size, comparisons between the low and high cell count assay sets are inconsistent, presumably since smaller populations are more easily resolved in the high cell count assay set. However, by considering the trends within each assay set, it can be seen that changes in sub-population size over time are generally less pronounced for the low O/N samples than they are for the high O/N samples. This is evident, in particular, from the low O/N samples in the centre of the bimodal region

[13] This will be explained in more detail in the following chapter.

Figure 5.11 (*Continued*): Two sets of short, normal and long equilibration time hysteresis assays of the ER-MC strain were measured by flow cytometry: the first set (upper panels) were measured for lower cell counts (10,000–35,000), and the second set (lower panels) were measured for higher cell counts (78,000–92,000). Indicated in italic text for each assay is the equilibration type (short, normal or long) and the recorded number of hours of equilibration time. The fluorescence distributions were curated as described in Figure 5.9 and the sub-populations are also plotted as described in that figure. The area of each point is proportional to the fraction of that sample that belongs to the respective component distribution. As in previous figures, colour is used to distinguish history (low/high overnight (O/N)) and presence or absence of Tum (Tum$^+$/Tum$^-$). Dotted lines mark the transitions between unimodal and bimodal descriptions of the samples.

where changes in sub-population size are comparatively static. For these samples, stochastic switching is likely to be close to equilibrium at the long equilibration time point, which further implicates sub-population size in those samples as a measure of the relative stability of each semi-stable state. In contrast, changes in sub-population sizes for the bimodal high O/N samples are much larger over time, and it is unlikely that stochastic switching in those samples has reached equilibrium by the long equilibration time point. As expected, the largest changes in population size (for both low and high O/Ns) are found for samples with CI production rates furthest from their initial states, indicating that those samples are located well outside the region of bistability.

5.5 Chapter Summary

Measurements of the Tum−CI MFL strains by flow cytometry provided a richer picture of its bistable behaviour than was obtained by LacZ assay. After filtering and normalisation for morphological differences, single-cell measurements of *pR* activity clearly revealed a bimodal population structure for samples in the presumptive bistable region of the MFL. In spite of substantial overlap of the sub-populations, a rigorous method for fitting a two-component mixture model to the data was devised to enable a quantitative analysis. The approach made use of the Tum$^-$ control distributions to construct a family of prototypical unimodal distributions based on empirically constrained skew-*t* distributions. These constrained skew-*t* distributions fitted the data extremely well and provided robust discrimination between unimodal and bimodal samples. The expected activity of each mixture model component compared favourably with the steady-state deterministic model for the long equilibration time hysteresis curves, and a semi-deterministic interpretation could be invoked to explain some basic trends in the data. Flow cytometry measurements made at multiple equilibration times further revealed that stochastic switching between the two stable states is an active process throughout the assay. In the coming chapter, a stochastic model of the MFL will be developed to enable a deeper investigation into the origins of this stochastic switching.

6

Investigating noisy switching in the Tum−CI MFL by stochastic modelling

The deterministic model of the Tum−CI MFL provided a simple analytical framework that indicated the potential for bistability in this gene circuit and motivated the hysteresis assay as an informative experimental test of its stability. Initial results obtained by LacZ assay presented only limited evidence for hysteresis of a whole population of cells. However, single-cell measurements by flow cytometry revealed that hysteresis had been obscured by noisy switching between two alternative stable states. Cell populations were in fact composed of two sub-populations, each with mean fluorescence intensities that appeared consistent with the steady-state deterministic model. Moreover, time-course measurements provided evidence for active switching between the two stable states over the course of the hysteresis assay. These results argued strongly for the generation of a significant level of noise at the single-cell level.

An important source of noise in bacterial gene networks is the stochasticity intrinsic to gene expression [McAdams and Arkin, 1997; Elowitz et al., 2002; Golding et al., 2005; Cai et al., 2006]. Such intrinsic noise arises in consequence of the random timing of biochemical reactions at the molecular scale and the finite, low copy numbers of network components such as promoter elements and mRNA [Ozbudak et al., 2002; Golding et al., 2005; Cai et al., 2006]. This results in 'noisy' protein production with significant cell-to-cell variation, which can cause average cell behaviour to deviate significantly from the equivalent deterministic trajectory. Indeed, gene networks that are analogous from a deterministic standpoint can exhibit distinct phenotypes dependent on their noise characteristics [Cağatay et al., 2009]. Far from being a hindrance, noise in gene networks has been observed to enable complex emergent behaviours like probabilistic sampling of alternative states [Losick and Desplan, 2008; Kittisopikul and Süel, 2010], adaptation on non-evolutionary timescales [Tsuru et al., 2011], and even bimodal cell populations [To and Maheshri, 2010]. A quantitative understanding of such behaviours requires stochastic modelling. Stochastic models recognise that molecules are most accurately represented as discrete quantities, not continuous concentrations, and that the seemingly random diffusion of molecules according to Brownian motion results in the random timing of molecular collisions and reactions [Gillespie, 1977]. As such, stochastic models of biochemical reac-

tion networks are necessarily probabilistic [McAdams and Arkin, 1997], characterising the time evolution of the probability distributions of each discrete species [Gillespie, 1992; Kepler and Elston, 2001].

This chapter begins with a brief introduction to these and other basic concepts of stochastic modelling and simulation. By considering the conditions under which a deterministic model is an appropriate approximation for a stochastic model, a hybrid stochastic/deterministic model is proposed for the Tum−CI MFL that is designed to capture the most important sources of noise. The model is deliberately kept simple to restrict focus to critical mechanisms and minimise the number of parameters. The hybrid model bears a number of parallels with the original deterministic model, and this is used to facilitate parameter estimation from the sub-population location data of the previous chapter. Parameters specific to the stochastic model are, however, chosen heuristically. The resulting model enables an exploratory analysis into the origins and roles of the noise generated in the Tum−CI MFL. Model simulations suggest that noise in CI production has a similar effect on the stability of low and high fluorescence populations, whilst noise in Tum production affects only the high fluorescence state. In order to reproduce the broad semi-stable region observed for the low O/N, the stochastic model requires large fluctuations in Tum production. These fluctuations are larger than would be expected simply from transcription and translation, implying that an important slow stochastic reaction, such as the partitioning of plasmids at cell division, is missing in the model.

6.1 Stochastic modelling of gene networks

Whilst more complex than their deterministic counterparts, stochastic models provide a more realistic description of gene network behaviour. Whereas a deterministic approach treats each species using continuous concentration variables whose time evolution is wholly predictable, the stochastic approach recognises that molecules and reactions are discrete and that the timing of reaction events is effectively random at the molecular scale. As such, stochastic models demand a probabilistic description: the time evolution of a species is modelled not as a representative average concentration, but as a time-dependent probability distribution that specifies probabilities for observing every possible state (i.e., every possible number of molecules). Like the ODEs that are specified for a deterministic model, a system of stochastic differential equations could, in theory, be specified for the time evolution of a joint probability distribution that simultaneously describes the probabilities for all possible states of all species in the network under consideration. Such a Chemical Master Equation (CME) can be rigorously derived using the fundamental principles of molecular kinetics [Gillespie, 1992], but its complexity prohibits exact solutions for any but the simplest cases. Instead, stochastic gene networks are typically studied by *simulating* trajectories that are consistent with the CME [Rao et al., 2002]. Inferences about

the probability distributions of the state variables are then derived by calculating statistics over multiple trajectories (typically 1000s).

The theoretical framework for simulating systems of stochastic reactions has been studied for many decades. However, on the back of experiments confirming and characterising the generation of noise in gene expression [Elowitz et al., 2002; Golding et al., 2005; Cai et al., 2006; Taniguchi et al., 2010], interest in stochastic modelling of gene networks has exploded over the last 15 years [Rao et al., 2002; Kaern et al., 2005; Paulsson, 2005; Wilkinson, 2009]. Many alternative stochastic simulation algorithms have been formulated to deal with the peculiarities of modelling gene networks; the most relevant are introduced and compared in the text by Wilkinson [2006]. However, by and large, these algorithms are all adaptations of the algorithm formulated by Gillespie [1977] and a basic understanding of Gillespie's algorithm is a useful prelude to the hybrid deterministic/stochastic model that is later introduced for modelling the Tum−CI MFL.

Gillespie's algorithm [Gillespie, 1977] is the canonical algorithm for simulating stochastic trajectories. The algorithm prescribes a Monte Carlo simulation method that generates exact realisations of the Markov process defined by the CME [Gillespie, 1992]. The foundation for the algorithm is a statistical mechanical argument (made precise in Gillespie [1992]) that, in a well mixed and thermally equilibrated container, the probability per unit time for a reaction between two freely diffusing molecules is constant in time. Put mathematically, the probability for that reaction to occur in some infinitessimal time interval $\mathrm{d}t$ can be written as $\lambda \mathrm{d}t$, where critically, the reaction 'hazard' λ has no probabilistic time dependence. Such a property is symptomatic of a homogeneous Poisson process — a memoryless process where the number of events counted over a time interval t follows a Poisson distribution $\mathrm{Pois}(\lambda t)$. However, the occurrence of the reaction events themselves make the overall stochastic process non-Poisson in general, since most reactions consume reagents or effect changes in reagent numbers for downstream reactions. Such events introduce a conditional (probabilistic) time dependence into the hazards. Gillespie's method circumvents this complexity by instead framing the problem step-wise in terms of times between events, which for a Poisson process are exponentially distributed. The algorithm proceeds iteratively over three steps: (1) randomly sampling a time to the first event using the combined hazard for all possible reactions, (2) randomly choosing which particular reaction will next occur as weighted by the individual reaction hazards, and (3) updating the state according to that reaction's stoichiometry and recalculating the hazards for the new state. In this way, the algorithm ensures that each probabilistic step involves no time-varying hazards by simply stepping forwards one reaction at a time, only updating the reaction hazards step-by-step. By repeatedly sampling the next reaction and stepping the system forwards, the desired simulation time can be reached.

To make the Gillespie algorithm more concrete, consider the hypothetical reaction $\mathrm{A} + \mathrm{B} \longrightarrow \mathrm{C}$. An example trajectory for this reaction is shown in Figure 6.1. Given N_A

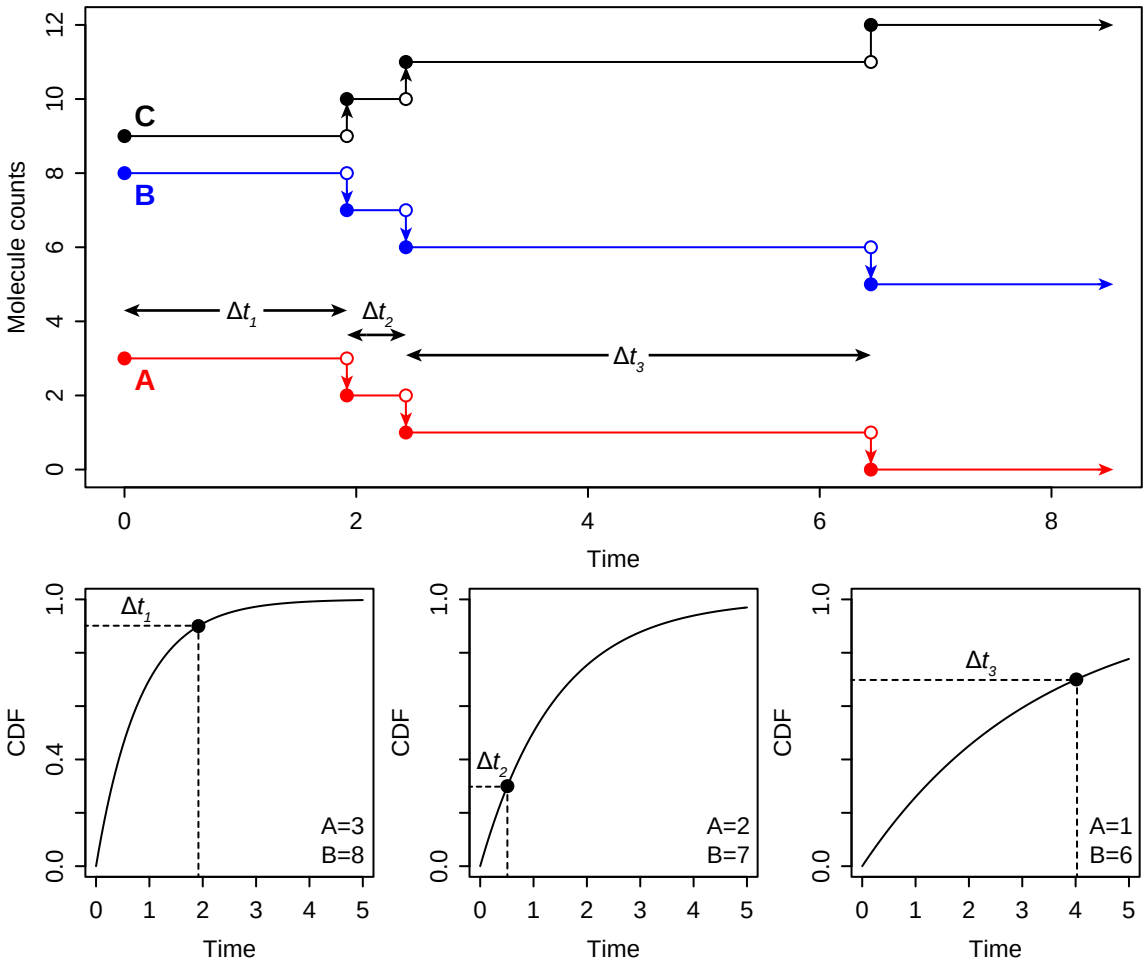


Figure 6.1: Illustrating the Gillespie algorithm. An example trajectory for the hypothetical reaction A + B ⟶ C is shown in the upper panel. The times between reactions are chosen by mapping random numbers from the unit uniform distribution ($U(0,1)$) the cumulative distribution function (CDF) of the exponential distributions shown in the lower panels. As each reaction occurs, the hazards change to reflect the new numbers of molecules, and this affects the shape of the CDF. In this case, A and B are consumed until one of them reaches zero. At that point the hazard becomes zero and there is no chance for the reaction to occur.

molecules of A and N_B molecules of B, the probability of *any* reaction occuring can be calculated by summing up the probability for a reaction between each A-B pair. Using basic combinatorics, the combined hazard can be derived, giving $\lambda = cN_AN_B$, where c is some reaction-specific rate constant that encapsulates both the probability for two molecules colliding and also the likelihood of that reaction given a collision. If this were a Poisson process, λ would be the average number of reactions occurring per unit time and times between events could be simulated by sampling from the exponential distribution, $\text{Exp}(\lambda)$. An exponentially distributed random variable T, written $T \sim \text{Exp}(\lambda)$, has cumulative distribution function (CDF) $F_T(t) = 1 - e^{-\lambda t}$. The CDFs for different hazards, λ, are shown in the lower panels of Figure 6.1. Times are sampled from these CDFs by mapping random numbers drawn from $X \sim U(0,1)$ — the unit uniform distribution — to the inverse of the CDF, $T \sim F_T^{-1}(X)$. That is, the time to the next reaction can be calculated

as[1] $t = -\log(u)/\lambda$, where u is randomly sampled from $U(0,1)$ so that the distribution of u is equivalent to $1-u$. Since by design, no other reactions can occur in this time step, the time is sampled exactly. After the reaction, however, the number of A and B molecules has decreased by one, and the new hazard becomes $\lambda = c(N_A - 1)(N_B - 1)$. With this new hazard, the time to the next event can now be generated randomly and the simulation again stepped forwards. Where there is more than one type of reaction, the time to the next event needs to be generated using the combined hazard for *any* reaction to occur.

Gillespie's algorithm, whilst exact, still demands too much detail to make for practical simulation of gene networks. The algorithm calls for the simulation of all fundamental diffusion-limited reactions, which would be a formidable task if each of the base pair transcription reactions, amino acid translation reactions or protein multimerisation steps were included. Such detailed models can be formulated, and have been for systems as complex as the phage lambda lysis-lysogeny decision [Arkin et al., 1998], but numerous parameter estimates are required and simulations consume significant compute time. It is more valuable in this context to keep the model simple and efficient. By capturing just the primary stochastic sources and approximating the details, parameter identifiability is maximised and simulation times greatly reduced. Though model integrity is lost, as long as the assumptions behind a simple model are well-understood, informative predictions and insights can still be made. To this end, the coming paragraphs discuss various approximations that can be made for stochastic models of gene networks. Firstly, conditions under which the discrete process can be treated as continuous are considered. Reactions best suited to the discrete regime should be maintained as such in the Gillespie algorithm, whereas a continuous process can, in many cases, be approximated deterministically. The processes of transcription and translation are identified as important sources of noise, and a simplified model of stochastic protein production is then introduced. Finally, an accurate way for incorporating deterministic reactions in the stochastic simulation algorithm is introduced.

A notable assumption of the deterministic model is that species can be treated as continuous concentrations. This greatly simplifies the mathematics, and motivates approximating the discrete stochastic process by a continuous one. The CME, or equivalently the Gillespie algorithm, requires a discrete formulation, but it can be shown that under certain conditions it can be approximated as a Chemical Langevin Equation (CLE)— a system of stochastic differential equations involving linear combinations of deterministic terms and (Gaussian) white noise terms [Gillespie, 2000; Rao et al., 2002; Wilkinson, 2009]. The CLE evolves as a continuous stochastic process, so that instead of jumping discretely, molecule numbers effectively diffuse over time. The CLE is a valid approximation when both (1) the rates of each reaction change slowly enough, and (2) the number of all stochastic reactions is sufficiently large on that time scale [Gillespie, 2000; Kaern et al., 2005]. These conditions

[1] Here, log is the natural logarithm; the base 10 logarithm is notated in this thesis as $\log_{10}$.

are typically satisfied when the counts of all involved molecules are high, though the CLE can provide accurate results for counts as low as tens of molecules [Grima et al., 2011]. Importantly, these conditions are satisfied for many of the high frequency reactions that might otherwise consume much simulation time.

Even more simplifications can be made where the mean behaviour of the CLE can be approximated as a deterministic trajectory. The CLE reduces exactly to a deterministic ODE when the variance of the noise terms is set to zero. However, if the variance relative to the mean is small but nonzero, the average behaviour of the system will often follow the deterministic trajectory. A well-known result is that the relative size of downstream fluctuations scales approximately as the inverse square root of the reactant populations [Gillespie, 2000], so deterministic behaviour is again favoured when population numbers are high. If the CLE contains only linear terms, it will follow the deterministic trajectory exactly [Wilkinson, 2006]. Since the noise terms are embedded in the CLE, however, for highly nonlinear systems the mean behaviour of the continuous stochastic process can diverge significantly from the deterministic trajectory, even for low levels of noise. This is most obviously the case when there are multiple stable states in the deterministic equations, since the system may with some (though perhaps low) probability switch into an alternative stable state [Gillespie, 2000]. In other cases, nonlinearity results in only minor deviations, and the simplicity brought by the deterministic equilibrium formulation makes it an attractive option even where some accuracy is lost. The high frequency and relatively fast dynamics of protein-DNA and protein-protein binding mean that such reactions can often be approximated as deterministic using equilibrium statistical mechanics [McAdams and Arkin, 1997]. Note that high molecule numbers alone do not guarantee the suitability of the approximation: slow rates of transcription factor unbinding could give rise to stochastic bimodality [Kaern et al., 2005].

Reactions dependent on species with low counts (e.g., promoters or mRNA) are likely to generate large relative fluctuations, so reactions like transcription and translation are typically best modelled stochastically [Thattai and van Oudenaarden, 2001; Ozbudak et al., 2002; Kaern et al., 2005]. Whilst there are many steps in the transcription and translation of a gene, it is the low-frequency rate-limiting steps that are the most influential in a stochastic model. A two-stage model of protein production has proven successful and popular in prokaryotes [Ozbudak et al., 2002; Swain et al., 2002; Friedman et al., 2006]. This simple model treats the transcription of mRNA as a single effective reaction, and the translation of protein as a first order reaction in the number of transcripts. Both mRNA and protein are also degraded in first order, so that the production of protein is dependent on the lifetime of the mRNA. If the half-life of the mRNA is shorter than the half-life of the proteins, stochastic gene expression can be further approximated as a single effective reaction in which proteins are produced in stochastic bursts that are exponentially distributed in time and geometrically distributed in size [Friedman et al., 2006; Shahrezaei and Swain, 2008].

The one-step protein production approximation relies on time-averaging over the fluctuations of the short-lived intermediate mRNA species, so that mRNA dynamics can be subsumed into an effective overall reaction. In spite of its simplicity, this one-step model has been shown to capture much of the essential variability intrinsic to gene expression in prokaryotes [Cai et al., 2006; Yu et al., 2006; Taniguchi et al., 2010]. By ignoring mRNA dynamics, this simple model produces simulations marked by instantaneous bursts of protein production. The size of the burst is geometrically distributed as a result of the exponentially distributed lifetime of an mRNA: assuming a constant rate of translation, the number of proteins produced from each transcript will be proportional to its lifetime, and the exponential distribution is the continuous equivalent of the discrete geometric distribution.

The one-step protein production reaction can be parameterised by a hazard, λ, that characterises the average rate of promoter firing, and a burst size, b, that gives the average number proteins produced per transcript. In relation to an overall rate of protein production, p, the hazard can be expressed as $\lambda = p/b$. This hazard can be used as per the Gillespie algorithm to simulate a time, Δt, to the next promoter firing event. Unlike in the Gillespie algorithm, however, the stoichiometry of the reaction is determined by randomly sampling a burst size from a geometric distribution. The total number of proteins, $N(t)$, then gets updated according to this sampled burst size:

$$N(t + \Delta t) = N(t) + \Delta N \tag{6.1}$$

where ΔN is sampled from $Geom(1/b)$, the geometric distribution with average burst size of b. This parameterisation has the added benefit that the burst size parameter, b, sets the magnitude of noise [Paulsson and Ehrenberg, 2000; Thattai and van Oudenaarden, 2001; Friedman et al., 2006]. This means that the burst size, b, can be chosen to produce the expected level of intrinsic noise for the production of a given protein. The rate of promoter firing, λ, is then determined using this burst size and the overall rate of production, p, which is closely related to deterministic production rates. Since transcription factor dynamics at the promoter are typically fast, it is possible to link the fractional occupancy of a transcription factor at its promoter to the rate of transcription, simply by scaling the hazard in the one-step protein production reaction. Note that even if transcription factor dynamics operate on the same timescale as mRNA, providing that the transcription factor off-rate is faster than protein decay, time-averaging can again be invoked to show that the one-step model continues to be appropriate[2] [Shahrezaei and Swain, 2008].

It can now be seen that most gene networks will consist of a number of important stochastic reactions, but also many reactions that could be treated more effectively as deterministic. This motivates a hybrid modelling approach where stochastic simulation is

[2] Note, however, that if one were instead following the distribution of mRNA, it would still be necessary to explicitly model the promoter state, as in Zong et al. [2010], since the timescale of transcription factor binding can be similar to the mRNA lifetime.

performed for selected reactions, with the others simulated by concurrently solving the (time-evolving) determininstic equations [Kiehl et al., 2004]. More accurate hybrid models can also simulate the additional noise that is produced from the high frequency equations by numerically simulating from their CLE [Salis and Kaznessis, 2005]. However, the simpler hybrid stochastic/deterministic model can still be used to accurately model the lysis/lysogeny decision in phage lambda [Kiehl et al., 2004], and is likely to remain suitable for studying stability in other bistable gene networks.

Two important extensions to the Gillespie algorithm are required for the hybrid stochastic/deterministic model: (1) maintaining well-synchronised discrete and continuous versions of each state variable, and (2) numerically integrating the deterministic equations between each stochastic event. The former can be addressed through a careful numerical implementation. The latter, however, deserves additional attention since the evolving deterministic equations can result in time dependent hazards [Haseltine and Rawlings, 2002; Alfonsi et al., 2005]. The introduction of a (deterministic) time-dependence into the hazards means that the stochastic reactions must be modelled according to an *inhomogeneous* Poisson process. To illustrate this in terms of the example reaction $\mathrm{A} + \mathrm{B} \longrightarrow \mathrm{C}$, consider the case where that reaction occurs inside a cell in an exponentially dividing culture. Cells in such a culture grow in volume, and this increase in volume can essentially be treated as continuous (see Wilkinson [2006] for a more thorough discussion of this example). In this case, the hazard $\lambda(t)$ now has a (deterministic) time dependence: $\lambda(t) = cN_AN_B/V(t)$ for some time-varying volume $V(t)$. By treating the hazard as constant over each infinitessimal increment $\mathrm{d}t$, then the number of events in that interval is approximately $\mathrm{Pois}(\lambda(t)\mathrm{d}t)$. Since the sum of Poisson distributions is also Poisson, this can be integrated to obtain a *cumulative* hazard $\Lambda(t)$:

$$\Lambda(t) = \int_0^t \lambda(s)\mathrm{d}s \tag{6.2}$$

where s is an integration variable. The number of events N_t occuring in time t is then given by the inhomogeneous Poisson process $N_t \sim Po(\Lambda(t))$. The time to the next reaction can be randomly sampled, by analogy with the sampling method for $\mathrm{Exp}(\lambda)$, as:

$$t = \Lambda^{-1}(-\log(u)) \tag{6.3}$$

where u is randomly generated from $U(0,1)$. Where a convenient analytical expression for the cumulative hazard $\Lambda(t)$ can be formed, the inverse can be directly calculated to obtain the above expression exactly. However, in more complex cases the integral must be numerically approximated to determine the time between steps. In that case, it is more convenient to calculate $\Lambda(t)$ directly and numerically integrate until it has a value matching $-\log(u)$. This method of numerical integration will be used for the stochastic model of the Tum−CI MFL that is introduced next.

6.2 A hybrid stochastic/deterministic model of the Tum−CI MFL

In order to investigate noisy switching between stable states in the Tum−CI MFL, a stochastic model must be derived. For the bistable *lac* operon, noisy switching during hysteresis has also been observed experimentally [Ozbudak et al., 2004], and in that case, a model based on the well-established deterministic equations, but including additional stochastic terms was sufficient to explain how the reporter distributions evolved over time [Mettetal et al., 2006]. By maintaining strong parallels with the deterministic model, Mettetal et al. were able to construct a simple but effective model where many of the required parameters were already known or could be easily derived from steady-state distributions. Given that the behaviour of CI−*pR* repression and much of the behaviour of the Tum−CI MFL can also be identified with the deterministic model, a similar approach to developing a stochastic model of the Tum−CI MFL is taken here. By identifying likely sources of internal noise in the Tum−CI MFL, a hybrid stochastic/deterministic model of the Tum−CI MFL is developed based on the deterministic model of Chapter 2. However, unlike the *lac* operon model, the hybrid model of the Tum−CI MFL used here includes time-varying hazards that require a more rigorous hybrid modelling approach. This section concludes with some example simulation time courses illustrating the main features of the model.

Like for the deterministic model, separation of reaction timescales can be exploited to reduce the stochastic model. As was mentioned back in Section 2.2.3, the rate of loss of a species effectively sets its timescale. The rates of loss of CI and GFP are likely to be amongst the slowest, and these are also particularly influential in setting the state of the switch. Loss of both these species is dilution limited, making these rates on the order of the cell generation time (around 30 minutes). In contrast, the rates of loss of mRNA and the dissociation rates of transcription factors are typically at least an order of magnitude faster. The rate of loss of mRNAs in the Tum−CI MFL is unknown, but in *E. coli*, mRNA half-lives are typically 3–8 minutes [Bernstein et al., 2002]. The *lac* repressor controls expression from the P_{lac} promoter with an average dissociation time of around 5 minutes [Hammar et al., 2014]. The CI repressor controls expression from the *pR* promoter, and since the *in vitro* binding strength of CI to its promoter (28 nM) [Dodd and Egan, 1996; Shearwin et al., 1998] is approximately three orders of magnitude weaker than that for the *lac* repressor [Bintu et al., 2005], dynamics of CI at the *pR* promoter are likely to be faster still. This means that fluctuations in promoter state and mRNA are fast relative to the dilution rate, and will make only time-averaged contributions to the distributions of CI and GFP. Such fast reactions can be treated under a quasi-steady-state approximation [Rao and Arkin, 2003] and subsumed into effective reactions [Shahrezaei and Swain, 2008]. This means that production of CI from P_{lac} and of GFP from *pR* can be simulated according to the one-step model introduced in the previous section, with equilibrium models used to scale the rates of promoter firing as in the deterministic model. The distributions of fluorescence activities in the Tum$^-$ control strains support such approximations, since they are unimodal

and their mean activities compare well with the determininistic model.

The rate of loss of Tum is less well defined, making it more difficult to choose an appropriate model for Tum production. Recall from the degradation rate measurements in Section 3.5.2 that a fraction of the available Tum protein is lost quickly with a half-life of 2–3 minutes, with the remainder seemingly lost only by dilution. Depending on whether the degradation-sensitive or degradation-resistant fraction is considered the active fraction, the timescale for Tum evolution could be close to that of mRNA/repressor binding, or alternatively dilution limited, as it is for CI and GFP. Interestingly, deterministic fits of the time-course LacZ assay data (Section 4.3.3) and also of the sub-populations determined by flow cytometry (to be introduced in Section 6.3.1) prefer a long half-life for Tum. Though this may be an artefact of fitting stochastic data with a deterministic model, it is tempting to assume in the first instance that Tum is lost only by dilution since it greatly simplifies the model. If active Tum does in fact equilibrate quickly, then both transcription factor binding dynamics and mRNA dynamics may need to be explicitly modelled, especially since stochastic reactions could be amplified through the feedback loop. On the other hand, since the *pR-tum* module is located on a multi-copy plasmid, averaging over the promoter states may, to some degree, compensate for such effects [Loinger and Biham, 2009]. To avoid these complexities at this stage, active Tum is assumed to be lost by dilution.

By making the assumption that Tum is dilution limited, the number of discrete species simulated in the stochastic model of the Tum−CI MFL can be limited to just the total levels of CI, Tum and GFP proteins. Transcription factor binding and mRNA levels can be treated under the quasi-steady-state approximation, as can formation and dissociation of the Tum−CI complex (recall from Section 2.4.1 that dissociation of the Tum−CI complex is likely to be fast). As such, the stochastic model should closely parallel a deterministic ODE specified in terms of total CI (C_{tot}), total Tum (T_{tot}), and total GFP (Z) concentrations, like that first introduced in Section 2.2.4:

$$\frac{\mathrm{d}C_{\text{tot}}}{\mathrm{d}t} = \mathrm{p}_C - \gamma \mathrm{C}_{\text{tot}} \tag{6.4a}$$

$$\frac{\mathrm{d}T_{\text{tot}}}{\mathrm{d}t} = \frac{\mathrm{p}_T}{1 + \left(\frac{C}{\varepsilon_R}\right)^{\mathrm{H}_R}} - \gamma T_{\text{tot}} \tag{6.4b}$$

$$\frac{\mathrm{d}Z}{\mathrm{d}t} = \frac{\mathrm{p}_Z}{1 + \left(\frac{C}{\varepsilon_R}\right)^{\mathrm{H}_R}} - \gamma Z \tag{6.4c}$$

Here targeted degradation of the complex has been neglected, and the rates of loss of all species have been set to the dilution rate. The analogous stochastic model has just three stochastic reactions corresponding to each of the production rate terms. Recall that the Tum−CI and CI−*pR* equilibria are encapsulated within the Tum and GFP production terms, which depend on both C_{tot} and T_{tot} via the concentration of free CI according to the

constraint equation:

$$C_{\text{tot}} = C + C\left(\frac{T_{\text{tot}} - 2C_{\text{tot}} + 2C}{\varepsilon_S}\right)^{\text{H}_S} \tag{6.5}$$

This equation can be solved for C using numerical root finding routines.

The dilution terms in Equation (6.4) will be modelled both continuously, with species concentrations reduced according to the expanding volume of the cell, but also stochastically, with species partitioned binomially upon cell division. Continuous volume expansion affects only species concentrations; discrete molecule numbers continue to increase until cell division. Species concentrations can be calculated at any given instant from the number of molecules of that species and the cell volume. Since volume expansion due to cell growth is linear [Kubitschek, 1990], it is treated as a continuous function of time relative to division and given by $V(t) = t/\tau_{\text{dbl}} + 1$, where τ_{dbl} is the cell doubling time and volumes are normalised to the cell volume just after division. When the relative time reaches τ_{dbl}, molecule numbers are randomly reduced according to a binomial distribution and the relative time is reset to 0.

Since the deterministic model was defined according to a constant-volume process, the time-averaged production rates specified in that model do not correspond directly to the hazards of the variable-volume stochastic model. Take, for example, a standard production/dilution process specified as $\frac{\text{d}C}{\text{d}t} = p - \gamma C$ in the constant-volume formulation. The equivalent variable-volume rate equation can be split into an equation specifying the rate of increase in the *average* number of molecules, $\frac{\text{d}N}{\text{d}t} = \lambda b$, and an equation specifying how the concentration decreases with increasing volume, $C(t) = N(t)/V(t)$. Using the quotient rule, the rate of change of the concentration can be expanded:

$$\frac{\text{d}C}{\text{d}t} = \frac{V\frac{\text{d}N}{\text{d}t} - N\frac{\text{d}V}{\text{d}t}}{V^2} = \frac{\lambda b}{V} - \frac{N}{\tau_{\text{dbl}}V^2} = \frac{\lambda b}{V} - \frac{C}{\tau_{\text{dbl}}V}$$

In order to reconcile the two formulations, the coefficients of C in the constant-volume equation must match to those in the variable-volume equation. Since cell volume changes periodically according to cell division, it is sufficient to ensure that the coefficients match over the period of one doubling time. For the production term:

$$p = \left\langle \frac{\lambda b}{V} \right\rangle_t = \frac{\lambda b}{\tau_{\text{dbl}}} \int_0^{\tau_{\text{dbl}}} \frac{1}{V}\text{d}t = \lambda b \int_1^2 \frac{1}{V}\text{d}V = \lambda b \log(2)$$

where the angular brackets denote a time average over the doubling period. The factor of $\log(2)$ gives the correct scaling between the two formulations. This means that in the stochastic model, the hazard for CI production can be expressed as $\lambda_C = \text{p}_C/\text{b}_C \log(2)$, where p_C is the rate of CI production in the deterministic model, and b_C is the average number of proteins produced per mRNA transcript. A notable omission in this formulation is gene replication, which may well stand as another important source of stochasticity. It would be relatively straightforwards to add replication events for each of the genes in

Table 6.1: Stochastic reactions in the Tum−CI MFL model. The cumulative hazards are used to simulate the time to the next reaction. The stoichiometries specify how species counts change as a result of that reaction. Here, each reaction produces a randomly-sized burst of protein.

Reaction	**Cumulative Hazard**	**Stoichiometry**
$\emptyset \longrightarrow \mathrm{CI}_2$	$\dfrac{\mathrm{p}_C t}{\mathrm{b}_C \log(2)}$	$\Delta \mathrm{C}_{\mathrm{tot}} \sim Geom(1/\mathrm{b}_C)$
$\emptyset \longrightarrow \mathrm{GFP}$	$\dfrac{\mathrm{p}_Z}{\mathrm{b}_Z \log(2)} \int_0^t \dfrac{1}{1+\left(\frac{C}{\varepsilon_R}\right)^{\mathrm{H}_R}} \mathrm{d}t$	$\Delta Z \sim Geom(1/\mathrm{b}_Z)$
$\emptyset \longrightarrow \mathrm{Tum}$	$\dfrac{\mathrm{p}_T}{\mathrm{b}_T \log(2)} \int_0^t \dfrac{1}{1+\left(\frac{C}{\varepsilon_R}\right)^{\mathrm{H}_R}} \mathrm{d}t$	$\Delta T_{\mathrm{tot}} \sim Geom(1/\mathrm{b}_T)$

the hybrid model, since the model already incorporates the cell cycle. However, in the interests of limiting the number of parameters in this exploratory model, such an extension will be left for future study.

The cumulative hazards for one-step stochastic production of each species can now be completely specified and are given in Table 6.1. These reactions are specified in terms of burst sizes, b_C, b_T and b_Z, and deterministic protein production rates, p_C, p_T and p_Z, for CI, Tum and GFP respectively. Since the hazard for production of CI includes no time dependence, it is a standard Gillespie reaction and the *cumulative* hazard (as specified in the table) increases linearly with time. The hazards for production of Tum and GFP, however, need to be treated according to an inhomogeneous Poisson process, since the level of repression changes over time according to the continuous loss of CI and Tum by dilution. Those cumulative hazards must be numerically integrated since the shared repression factor is dependent on the level of free CI, which is itself determined by numerical solutions to the constraint equation, Equation (6.5). As per the one-step model of production, the result of any of these stochastic reactions is to increase protein levels according to a geometric distribution with mean given by the respective burst size parameters.

Custom software was written in C^{++} to implement the hybrid stochastic/deterministic model of the Tum−CI MFL (and similarly a model of the Tum$^-$ controls). The hybrid simulator uses the Next Reaction variant [Gibson and Bruck, 2000] of the Gillespie algorithm and is based on algorithms described elsewhere [Kiehl et al., 2004; Alfonsi et al., 2005], with hazards integrated according to an Euler time step. The state of the system can be completely characterised from the current time, t, the current volume, V, and three continuous state variables giving the total concentrations of Tum, CI and GFP. Protein numbers can be calculated using the product of concentration and volume and are made discrete by rounding down to the nearest integer. Only the three burst size parameters are specific to the stochastic model; the remaining parameters relate directly to the deterministic

model. Further details of the stochastic model and excerpts of the code are provided in Appendix D.

To illustrate key features of the stochastic model and simulation method, some sample time courses from the Tum^- control model are shown in Figure 6.2. The Tum^- model is identical to that of the Tum^+ strain except that all Tum evolution terms are omitted. The activity of the P_{lac} promoter (p_C), shown in the lowermost plots, is set to mimic the hysteresis assay. Time courses start from either a simulated low O/N (left) or high O/N (right) culture, and then at time $t = 0$ the activity of P_{lac} is set to an intermediate level of induction and simulation continued for the desired equilibration time. In the figure, the final induction levels were chosen to be far from initial levels to show the effect of a large change in induction conditions. Since the stochastic model produces Monte Carlo simulations, five samples of each trajectory are shown. An additional random time of simulation in O/N conditions is performed prior to collecting data to ensure that the timing of cell division is different for each trajectory. Species concentrations are favoured over protein numbers, because concentrations are measured in the *pR* activity assays[3] and concentration measurements are easier to relate to the deterministic model. For reference, deterministic trajectories are shown overlaid on the stochastic ones. Since the Tum^- controls have a single stable state, and since large relative changes in protein levels require many promoter firing events, it is not surprising that the average behaviour of the stochastic trajectories tends to follow the deterministic curves. Individual stochastic trajectories can, however, diverge significantly from the deterministic one, and this indicates the level of noise around the average trajectory.

The combined effects of stochastic one-step protein production and continuous dilution can be seen in the individual stochastic trajectories. The one-step model of production results in instantaneous bursts of protein production, which is particularly obvious for the large bursts. These randomly-sized bursts are immediately followed by hyperbolic loss in species concentrations according to the rate of increase in cell volume. This bursting is symptomatic of the one-step model of production, since it is assumed that mRNA is very short-lived — effectively lost as soon as it is produced — so that the dynamics of protein concentrations are dominated by promoter firing and dilution. The hyperbolic decrease in protein concentrations due to dilution becomes exponential as a result of cell division. This effect can be seen particularly in the decay of GFP protein from the high activity state (left panel), and to a lesser degree in the decay of CI protein from the high induction state (right panel).

The behaviour of the stochastic model is better resolved when considering many thousands of trajectories. The histograms in the upper panels of Figure 6.2 show the distribution of *pR* activities at time $t = 200$ minutes for 5000 sample trajectories. Consistent

[3] Normalisation by morphology (Section 5.1.4) in the flow cytometry assays effectively turns the measured absolute level of fluorescence into a level of fluorescence per the average cell.

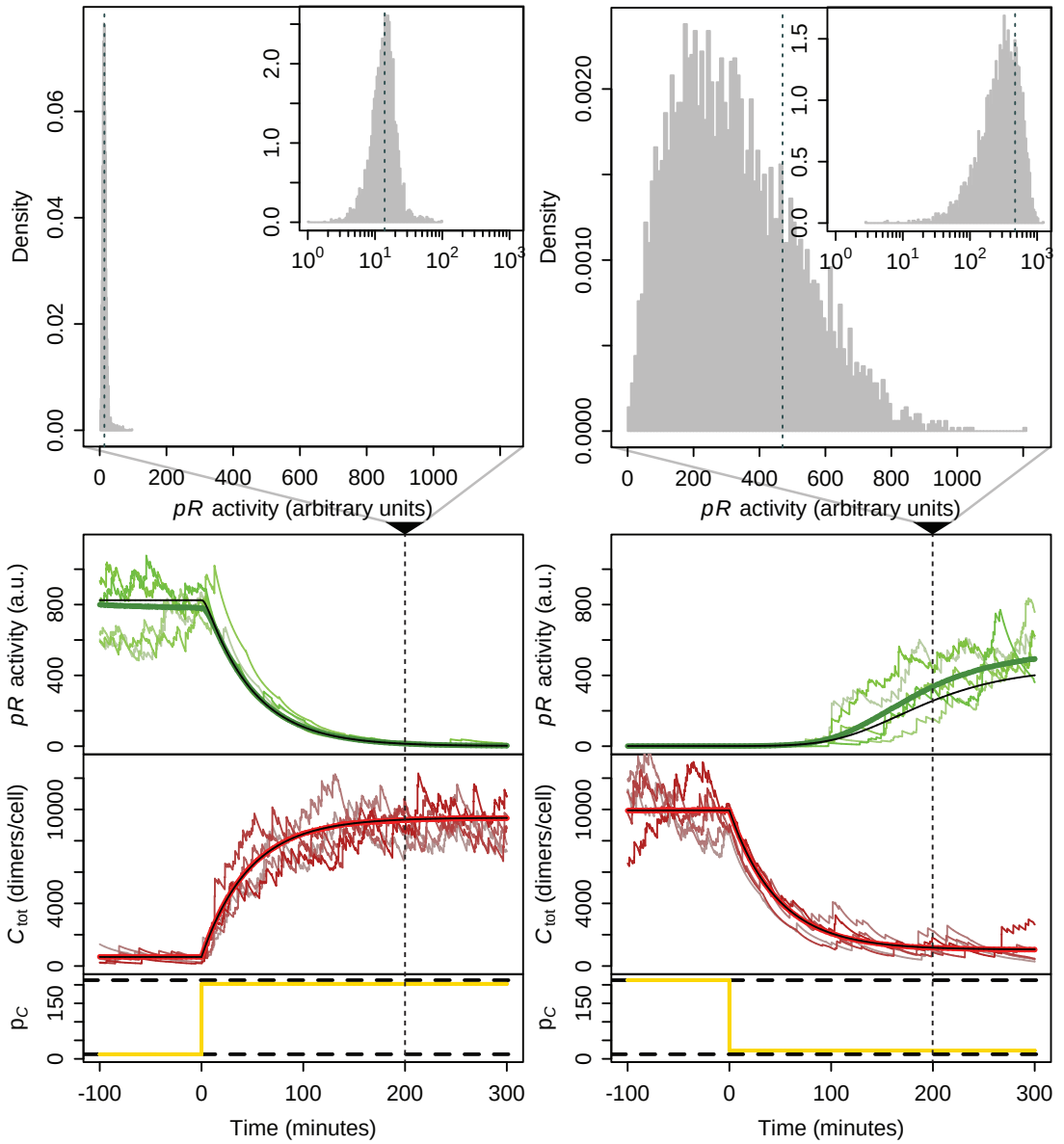


Figure 6.2: Noisiness in the hybrid stochastic/deterministic model of the Tum$^-$ control strain. Two conditions are tested: the left panels show simulations of the transition from a low O/N culture to a high CI production rate; the right panels show instead the transition from a high O/N culture to a low CI production rate. The variation over time of P_{lac} promoter activity (p_C), total CI concentration (C_{tot}) and *pR* activity are shown in the lower plots: five individual stochastic trajectories are shown for each initial condition and are distinguished by shading; thick solid lines show the average of 5000 stochastic trajectories; thin black lines show the corresponding deterministic trajectory. In the plot of P_{lac}, thick black dashed lines indicate the activities ascribed to low and high O/N cultures. The upper plots show histograms for the distribution of simulated *pR* activities of 5000 trajectories measured at time $t = 200$; insets show the same distribution on a logarithmic scale to facilitate comparisons with the flow cytometry results; vertical dotted lines show the value predicted deterministically. Parameters for the models are as listed in the 'Simple fit' column of Table 6.2, with the burst sizes of CI and GFP defined to be 300 dimers and 20 arbitrary units (a.u.) respectively.

with theory [Gillespie, 2000], noise levels (the widths of the distributions) increase with increasing protein levels (activity). For the burst-size parameters used in this example, the distribution with high activity (right panel) has a width that is comparable with those of the flow cytometry samples (compare the logarithmic inset with the histograms in Figure 5.5, for example). The activity predicted according to the deterministic model does not necessarily overlap with the average behaviour. In the low activity state, the line marking the deterministic activity is close to the centre of the distribution, whereas it is off-centre for the high activity state. This likely reflects how the propagation of noise through the nonlinear CI$-pR$ equilibrium distorts the macroscopic trajectory relative to the deterministic model.

6.3 Establishing a parameter regime for the hybrid stochastic/deterministic model

The hybrid stochastic/deterministic model of the Tum$-$CI MFL introduces only three additional parameters compared with the deterministic model. These additional parameters are the burst sizes for production of each of the MFL proteins (CI, Tum and GFP), and can be used to tune the level of intrinsic noise generated in the circuit. Direct measurements of these parameters are unavailable at this time so suitable estimates will need to be made. Estimates for the deterministic parameters also need to be revised. Though many of these were measured experimentally as reported in Chapter 3, the measurements were weakly constrained and result in model curves giving a poor match to the flow cytometry data sets. Additionally, the parameters determined in Section 4.3 by fitting the deterministic model to LacZ assay data are unlikely to be accurate, since the flow cytometry results show that the average trajectory of pR activity is biased by stochastic switching between the stable states. Ideally, the deterministic and stochastic parameters would now be determined by fitting the hybrid stochastic/deterministic model to the flow cytometry data. Unfortunately, stochastic models pose additional analytical challenges: not only do they involve more parameters, but solutions are substantially more difficult and time consuming to obtain, putting model-based statistical inference out of reach in many cases[4]. Instead, the task of parameter estimation is split into an improved method for statistical inference of the deterministic parameters using the deterministic model, and a heuristic search for suitable stochastic parameters using the stochastic model.

6.3.1 Fitting the dynamic deterministic model to a stochastic data set

In spite of the inherent stochasticity of the Tum$-$CI MFL, it was previously maintained in Section 5.3 that a mixture model analysis of the flow cytometry data provided a means

[4] The necessary groundwork for fitting stochastic models to data is, however, in active development. See Wilkinson [2009]; Henderson et al. [2009]; Munsky et al. [2012]; Neuert et al. [2013].

for recovering deterministic behaviours. Fitted sub-populations were interpreted as the projection of a bimodal population onto deterministic components, and these were found to compare more favourably with the steady-state deterministic model. Such a deconvolution of the data opens the possibility for estimating the deterministic parameters from the data independently of the stochastic model. The manual choice of parameters for the steadystate model seen previously in Figure 5.10 stands as a promising starting point. However, with time-course data also available, an extension to the dynamic regime and an automated method for fitting the parameters would be preferable. In this section, by first revisiting the conditions under which deterministic approximations to stochastic models are accurate, a rudimentary method is proposed for reducing the time-course data to a presumptive deterministic trajectory. The deterministic model can fit this trajectory very well, but only if unrealistic parameters are allowed. By appealing to a stochastic explanation for that concession, a revision of the trajectory is proposed. More realistic parameters are then obtained by using a simplified parameterisation and favouring the long equilibration time assay.

The Tum$^-$ strains show monostable behaviour that is reasonably well-described by the determininistic model. As became evident from the sample trajectories in Figure 6.2, the stochastic trajectories of the Tum$^-$ controls by and large follow the deterministic one. This makes sense from a theoretical point of view: the Tum$^-$ control circuit stands as good candidate for modelling by the CLE (a continuous stochastic process), bringing it closer to the deterministic trajectory. Recall that a CLE approximation to the discrete stochastic process requires that the least frequent reactions remain frequent enough that appreciable changes in reaction propensities require many reactions. Here, where the dilution rate sets the speed of equilibration of both CI and GFP, the P_{lac} and pR promoters only need to fire many times each cell division. This is likely to be the case except where either of the promoters are close to fully repressed. Discrete production of GFP from a repressed pR promoter can be ignored, since it has no downstream effects and produces fluorescence indistinguishable from the background. Discrete production of CI, however, deserves some attention.

The ultrasensitive CI−pR equilibrium partially protects against discrete effects due to low molecule numbers, but concomitantly reduces the efficacy of the deterministic approximation. When CI numbers are low, that is, in uninduced cultures where P_{lac} is fully repressed, the CLE approximation would typically fall short. However, in this case, a crude argument can still be made for the approximation: low numbers of CI are buffered by the CI−pR equilibrium, such that appreciable changes in downstream reaction propensities (i.e., in the rate of pR firing) only occur for higher levels of CI. Unfortunately, the nonlinearity also means that a further approximation of the continuous stochastic process as a deterministic one suffers reduced accuracy. Nonlinear propagation of noise through the CI−pR equilibrium will asymmetrically bias the average behaviour. This means that a

fitted Hill coefficient will be a biased estimate that does not relate directly to the noise-free Hill parameter that is shared between the controls and the MFL. This effect is evident in the right panel of Figure 6.2, where the mean stochastic trajectory and deterministic trajectory are seen to diverge. Here this discrepancy is assumed to be sufficiently negligible that core behaviour is nonetheless retained.

In the full Tum−CI MFL gene circuit, the Tum−CI and CI−*pR* equilibria present complementary ultrasensitive hurdles to low levels of Tum or CI, respectively. A rough argument can be used to again validate use of the CLE approximation: in order to overcome the ultrasensitive barrier between the two zones of stability, numerous firings from either the P_{lac} or *pR* promoters are likely to be required. In other words, 'appreciable' changes in system state are likely to require many stochastic reactions. However, a deterministic approximation to such a CLE would be less apt: switching between the two stable regions of the CLE cannot be captured in the deterministic model [Gillespie, 2000]. Nonetheless, if the timescale of switching between the two basins of stability is slower than the rate of equilibration local to each basin, then a local treatment of the CLE (a diffusion process) can be applied within each basin [van Kampen, 2007] and the rate of switching between the two treated as a first exit problem [Aurell and Sneppen, 2002]. The behaviour local to each stable basin can then be followed using the mixture model fitted to each fluorescence distribution. The fact that the Tum$^-$ controls at each time point provide apt templates for the local fluorescence distributions can be attributed to the difference between the two circuits being essentially just the addition of high frequency reactions from the Tum−CI equilibrium[5]. Such high frequency reactions contribute relatively little additional noise, so that the shapes of the local distributions are still largely determined by the noise characteristics of production from P_{lac} and *pR*. This means that, like in Section 5.3, it can be assumed that the family of skew-*t* distributions defined by the Tum$^-$ control distributions give the prototypical stochastic spread of the regulated *pR* promoter around some deterministic trajectory. The fitted mixture model components of the Tum$^+$ MFL samples can then be interpreted as independent sub-populations with a locally valid deterministic average value.

It is well established that stochastic gene network models can extend effective bifurcation boundaries beyond those of their deterministic counterparts [Kepler and Elston, 2001]. In terms of the low O/N MFL samples, this effect might be understood as identifying the high intensity sub-populations at high levels of induction as stochastic — trapped in the high intensity stable state according to some stochastic trajectory. In that case, the point of bifurcation for the equivalent deterministic model would occur at some point in

[5] It is assumed that the production of Tum from the pMTS-*pR-tum*$^+$ plasmid has noise characteristics that satisfy the conditions of a continuous stochastic process like the output of the (pIT-CH-*pR-gfp*)$_{\text{HK}}$ module seemed to. With more copies of the *pR-tum* module, more frequent firing of the *pR* promoter is expected; the unknown is how the stochastic copy-number of the plasmids might affect the resulting distribution. This will be further discussed alongside the results of the stochastic model presented later in the chapter.

Table 6.2: A comparison of experimental parameter estimates and the deterministic parameters fitted to the time-course flow cytometry data measured for the ER-MCTum−CI MFL. The upper block contains the core MFL parameters; the middle block contains the initial condition scaling parameters; the lower block contains the timing parameters. Unless otherwise indicated, limits specify 95% confidence intervals.

Parameter	**Experimental**[*]	**Full fit**[**]	**Simple fit**[†]
ε_R (CI dimers/cell)	$(2.5 \pm 0.8) \times 10^2$ [‡]	9.39×10^2	9.74×10^2
H_R	2.2 ± 0.2 [‡]	3.05	3.01
ε_S (Tum monomers/cell)	$(1.28 \pm 0.04) \times 10^3$ [§]	7.98×10^3	2.67×10^3
H_S	4.6 ± 0.6 [§]	1.19	1.04
M_T (Tum monomers/cell)	$(2.5 \pm 0.7) \times 10^4$	3.19×10^4	3.18×10^4
Doubling time (min.)	28.2 ± 0.2 [¶]	27.1	32.4
Tum half life (min.)	2.7 ± 0.2	153.5	32.4
Low O/N scaling	–	4.49	–
High O/N scaling	–	1.67	–
Initial Tum scaling	–	7.86	–
Assay time (hours)[#]			
LCC, short equil.	4.9	2.4	–
LCC, normal equil.	6.2	3.7	–
LCC, long equil.	7.2	4.7	–
HCC, short equil.	4.1	3.0	–
HCC, normal equil.	5.6	4.5	–
HCC, long equil.	6.6	5.6	5.0

[*] Parameters determined experimentally in this thesis or derived from the literature.

[**] Parameters from the model shown in Figure 5.11 fitting all available parameters to match the time-course data.

[†] Parameters from the model shown in Figure 6.3 fitted to match a single long equilibration time assay with a simplified parameter scheme.

[‡] Parameters as fit by Dodd et al. [2007b]; limits indicate 'acceptable ranges'.

[§] Obtained by fitting *in vitro* gel shift data of Shearwin et al. [1998] to the model described in Section 2.2.2; ε_S was further converted to a number of monomers per cell using the scaling factor from Section 3.4.4.

[¶] The mean doubling time of all MFL strains (Tum^+ and Tum^-) with a medium copy *pR-tum* plasmid.

[#] The period of log phase growth attributed to each of the Low Cell Count (LCC) and High Cell Count (HCC) short, normal and long equilibration time (equil.) assays.

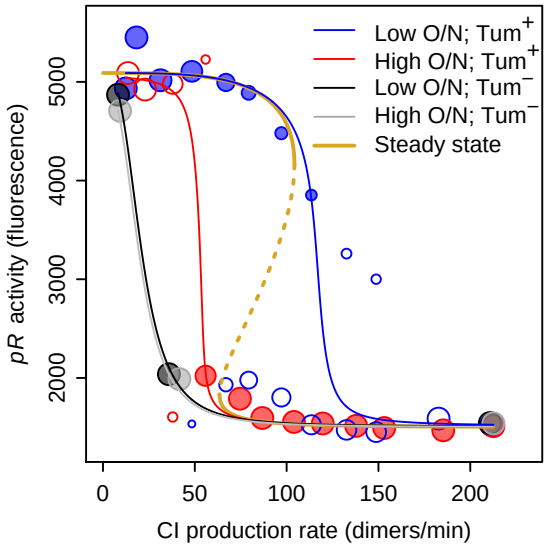


Figure 6.3: The deterministic model matches the long equilibration time data in spite of a simplified parameter set. The high cell count long equilibration time data was fit using a conservative set of parameters as described in the text. The data is the same as that in the lower rightmost panel of Figure 5.11, but, prompted by concerns that sub-populations near the bimodal/unimodal boundaries represented stochastic trapping in the initial state, two high intensity sub-populations in the low O/N Tum$^+$ samples were additionally excluded. Shown overlaid are the simulated deterministic time-course hysteresis curves for both strains. The equivalent steady-state curve for the Tum$^+$ strain is also shown; solid lines indicate stable states and dotted lines indicate the unstable states.

the middle of the bimodal samples, as would the turning point for the dynamic deterministic trajectory. The deterministic model does not account for such a stochastic effect, so in fitting the bimodal data with a deterministic model, there remains a source of uncertainty in identifying which of the sub-populations is more closely linked with the deterministic trajectory versus those better linked to stochastic switching. To this end, an *a priori* assignment of presumed deterministic sub-populations was trialled. This trial is presented in more detail in Appendix C.3, and though surprisingly good matches to the data could be found, attempts to simultaneously fit the complete data collection (i.e., all equilibration times and both low and high cell count data sets) consistently produced parameters that were unphysical. The best set of fitted parameters from this trial are shown in the 'Full fit' column of Table 6.2. As can be seen from the table, the good match of the data came at the cost of permitting the Tum half-life to be over five times longer than the fitted cell doubling time. It was found that this unphysical value was necessary to accommodate the slow decline of the low O/N hysteresis curve. One remedy would be to revise the assignments of the sub-populations. However, given the ambiguity in this decision, an alternative approach was pursued.

The limitations of the deterministic model are more apparent at earlier time points, when the system is further from equilibrium. The long equilibration time assays are the closest to steady-state, which also makes a deterministic description of these assays more likely to be reliable. As such, the model parameters from the 'Full fit' column of Table 6.2 were further optimised to fit just the high cell count long equilibration time assay, which is shown with the fitted model in Figure 6.3. Given that populations near the switching boundaries were likely to be stochastically trapped near the initial state and not actually deterministic populations, a few additional data points at the outer boundaries of the bimodal region were excluded; points excluded from the fit are indicated by open circles in the figure. Though the starting parameters for this final optimisation run were those

from the 'Full fit' column of Table 6.2, a number of additional parameter constraints were imposed to simplify the model. These simplifications were made under the premise that any extensions to the model that had been made to improve fits to the data were likely compensating for what were in fact stochastic effects. So the three O/N scaling parameters that were introduced to account for some perceived lag time were all set to one in favour of a slightly reduced equilibration time (5 hours), and the rate of loss of Tum was constrained to match the dilution rate like for CI and GFP. The optimised parameters can be found in the 'Simple fit' column of Table 6.2. Most of the core parameters are very similar to the starting parameters, with the notable exceptions being the doubling time, Tum half life and EC_{50} for the Tum−CI equilibrium (ε_S). With the deterministic model still fitting most of the sub-population data very well, and with all parameters now within realistic ranges, the 'Simple fit' parameters stand as the most suitable candidates to define the deterministic component of the hybrid stochastic model.

6.3.2 Optimising the magnitude of noise to reproduce observed rates of stochastic switching

With the deterministic parameters now constrained through fits of the sub-population means, it remains to estimate values for the stochastic parameters of the hybrid stochastic/deterministic model. These parameters are the average burst sizes for production of CI, Tum and GFP. The average burst size derives from the rate of protein translation and the average lifetime of the mRNA. None of these parameters have been measured for transcripts in the Tum−CI MFL, and extrapolation from measurements obtained elsewhere would give rough agreement with the experimental results at best. However, with only three parameters, a limited heuristic search of the parameter space is feasible. In this section, simulations using a range of burst sizes are presented and compared, and the final parameters chosen to optimise rates of stochastic switching and to give qualitative agreement with the distribution shapes observed by flow cytometry.

The choice of average burst size for protein production has a direct effect on the variance of the resulting protein distribution. A burst size parameter, b, sets the variance in protein numbers, σ^2, according to

$$\sigma^2 \simeq \mu(b+1) \tag{6.6}$$

where μ is the mean number of proteins at steady state [Paulsson and Ehrenberg, 2000; Thattai and van Oudenaarden, 2001]. In this way, the average burst size scales the magnitude of intrinsic noise for some given μ as determined by the rate of production. The widths of the unimodal Tum^- control distributions measured by flow cytometry give some indication of the intrinsic noise of GFP production. Preliminary simulations of the Tum^- control circuit revealed that the burst size of CI had only a small effect on the resulting downstream variance in the GFP distribution. So the burst size for production of

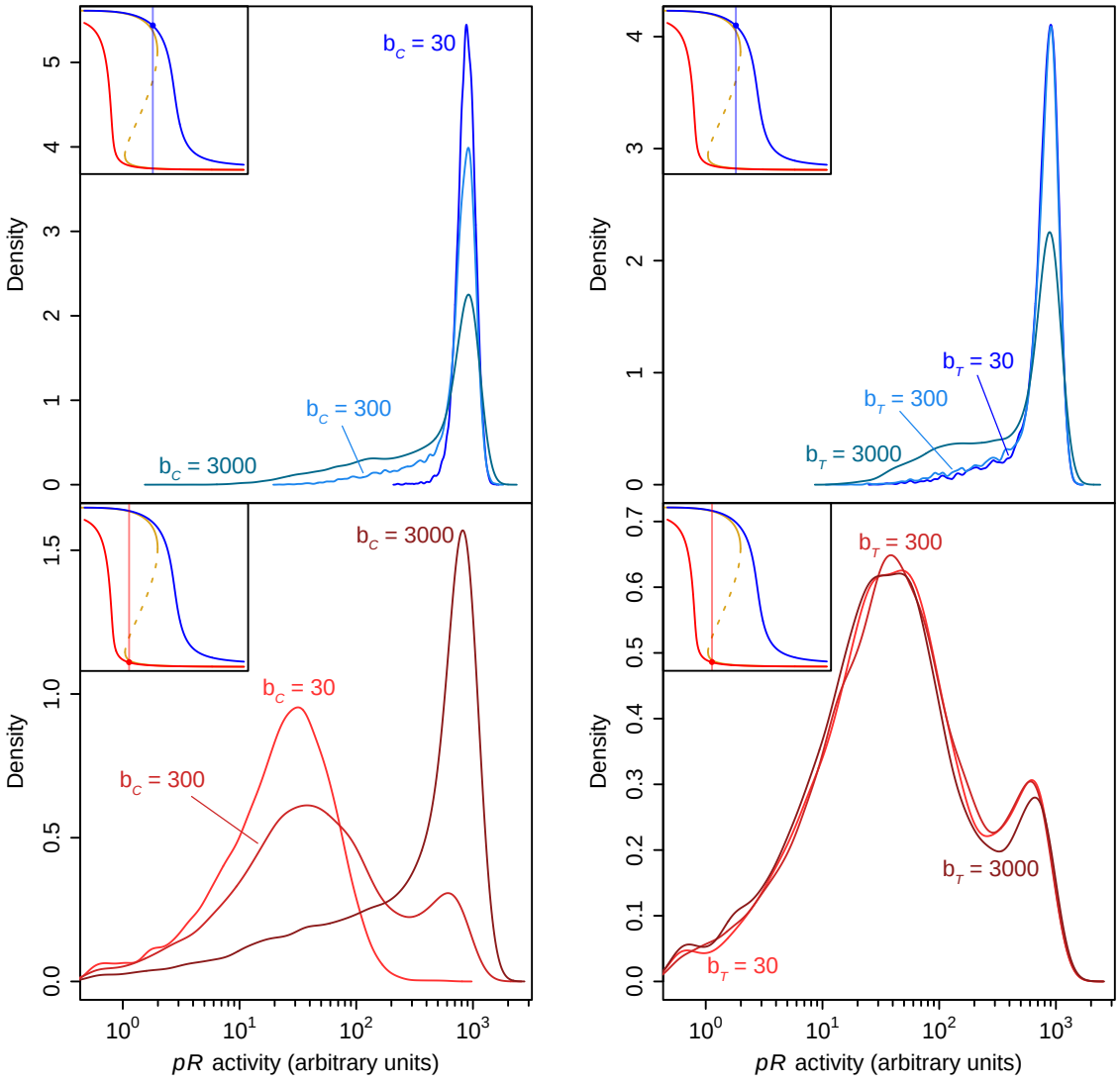
GFP could be independently optimised to match observed distribution widths. Relative to the nominal maximum reporter activity of $M_Z = 1000$ arbitrary units, a burst size of 20 units per transcript was found to produce simulated distributions that would compare well with the experimental distributions across a range of activity levels.

The burst sizes for production of CI and Tum were chosen based on how each parameter affected rates of stochastic switching. Average burst sizes in *E. coli* can range from one protein to many hundreds of proteins per mRNA transcript [Taniguchi et al., 2010]. A selection of burst sizes chosen from within this range were trialled in simulations of the Tum−CI MFL. The resulting distributions of reporter activities at 220 minutes of simulation time were examined for two alternative points in the hysteresis curve and are shown in Figure 6.4. These alternative simulation points were chosen to be at P_{lac} induction levels where stochastic switching was likely to occur, but to still be located within the region of deterministic bistability. Since protein distribution widths, i.e., their standard deviations, have a square root dependence on the burst size, the selection of burst sizes was varied by order of magnitude (i.e., 30, 300, or 3000 dimers or monomers per mRNA).

As can be seen in Figure 6.4(a), increasing the magnitude of noise in CI production reduces stability of both low and high fluorescence stable states. The upper plot shows that simulations starting near the low fluorescence stable state produce a unimodal peak centered over the initial state when the burst size is small ($b_C = 30$ dimers). However, when the burst size increases to $b_C = 300$ dimers, a bimodal distribution is observed, indicating that many of the stochastic trajectories have switched to a high intensity stable state. The upper plot shows the converse case: simulations starting near the high intensity stable state produce a sharp unimodal peak when $b_C = 30$ dimers, but as burst sizes increase to $b_C = 3000$ dimers, a lower intensity sub-population starts to appear. At least for the two hysteresis conditions shown, the burst size of CI production seems to have a much greater effect on the relative stability of the low intensity stable state.

As can be seen in Figure 6.4(b), increasing the magnitude of noise in Tum production reduces stability of the Tum−CI MFL only for the high fluorescence state. In the upper plot, the effects of Tum burst size on the high intensity stable state are shown. An increase in the number of low-fluorescence trajectories is only evident with the (overly) high burst size of $b_T = 3000$ monomers. However, this increase in switching was larger than that observed for the equivalent increase in CI burst size. In contrast, the lower plot shows that, regardless of Tum burst size, essentially no change in the extent of switching from low- to high-fluorescence states occurs.

These simulations reveal that the relative stabilities of each stable state can be independently optimised using the CI and Tum burst size parameters. The CI burst size can be increased to favour switching primarily out of the low fluorescence stable state, whereas the Tum burst size can be increased to favour switching out of only the high fluorescence stable state.



(a) Varying CI burst size

(b) Varying Tum burst size

Figure 6.4: The level of noise in CI production affects rates of stochastic switching differently to that for Tum production. **(a)** Increasing the burst size for CI production decreases stability of both the low and high fluorescence sub-populations. **(b)** Increasing the burst size for Tum production decreases stability of only the high fluorescence sub-populations. Shown are kernel density estimates of the population statistics for 10000 trajectories of the Tum−CI MFL stochastic model at 220 minutes. Insets indicate history and final CI production rates (68.48 dimers/s for the low O/N cultures; 98.5 dimers/s for the high O/N cultures) on the deterministic time-course and steady-state curves from Figure 6.3. Deterministic parameters are as listed in the 'Simple fit' column of Table 6.2. Stochastic parameters default to $\mathrm{b}_C = 300$ dimers, $\mathrm{b}_T = 300$ monomers and $\mathrm{b}_Z = 20$ units for CI, Tum and GFP burst sizes respectively; variations from the default are annotated for each curve.

Burst sizes of 3000 proteins per mRNA are outside of the range of measured burst sizes [Taniguchi et al., 2010], but these were included because the smaller burst sizes were unable to replicate the switching observed experimentally. In particular, the low O/N simulations (upper panels of Figure 6.4) did not produce a noticeable low intensity sub-population until the burst sizes were very large. In contrast, the experimental data shown back in Figure 5.8 for a low O/N measured at 60 μM IPTG distinctly shows two sub-populations. Furthermore, that sample is for a CI production rate *lower* than that simulated in Figure 6.4; that is, switching to the low intensity sub-population in the simulation shown here should be more frequent than that in the given experimental example. In order to replicate this rate of switching and since Tum burst size affects only the stability of the high intensity sub-population, the burst size parameters for CI and Tum were fixed to $b_C = 300$ dimers and $b_T = 3000$ monomers respectively for subsequent simulations. Though this would be an unphysical choice for b_T if it was interpreted literally as an mRNA burst size, it is still used here under the assumption that this parameter also encapsulates some other source of significant stochastic variation. Alternative sources of noise that might explain this requirement will be discussed towards the end of the chapter in Section 6.5.

6.4 A simple stochastic model is sufficient to reproduce salient features of the Tum−CI MFL

Using the parameters determined in Section 6.3, the hybrid stochastic/deterministic model of the Tum−CI MFL can be interrogated both to confirm that it reproduces the salient experimental results, but also to infer mechanism from details not captured in the experiments. In this section, model simulations are used to show that the simple stochastic Tum−CI MFL model that has been described produces bimodal populations at the expected points in the hysteresis curve. By then considering sample time-course trajectories, the stochastic events leading to switching between the states are examined. Finally, simulations ending at different equilibration times are analysed using the mixture modelling methods of Section 5.2. Whilst some discrepancies are clear, most of the qualitative behaviours observed experimentally can also be seen in the modelling results. A comparison of these results with the determistic model highlights the minor quantitative differences seen between the stochastic model and the interpretation of the sub-populations as deterministic.

To examine the behaviour of the stochastic model around the hysteresis loop, hysteresis simulations were performed using the Tum−CI MFL model for a range of final P_{lac} induction levels. These Monte Carlo simulations were repeated 5000 times for each final level of induction, and the distribution of *pR* activities recorded at time $t = 220$ minutes after the switch from O/N to final levels of induction. Two examples of these distributions are presented as histograms in the upper panels of Figure 6.5. The examples are for final

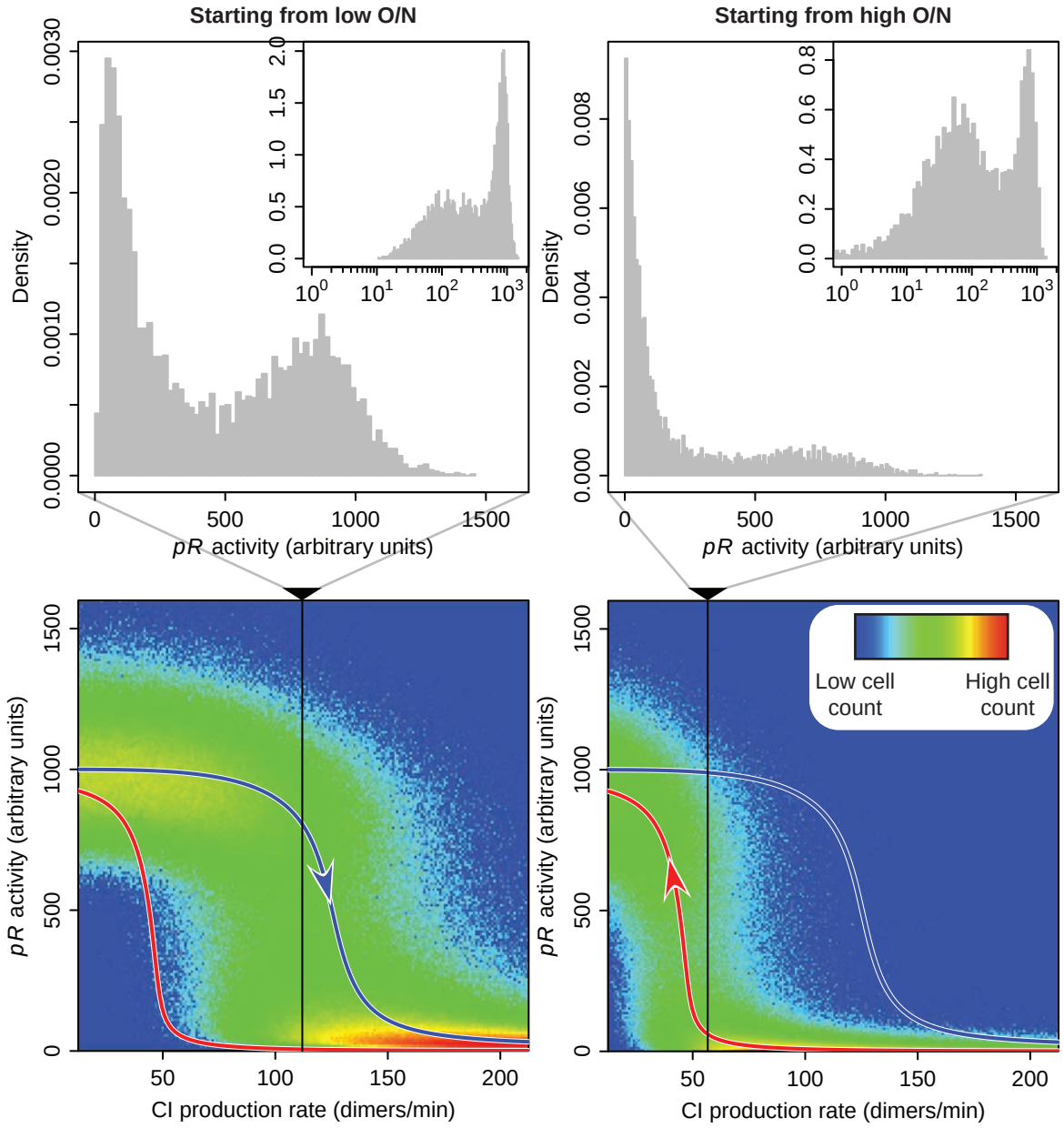


Figure 6.5: The hybrid stochastic/deterministic model reproduces the broad regions of bimodality observed experimentally. The distribution of *pR* activities at time $t = 220$ minutes was recorded for 5000 simulated hysteresis trajectories of the Tum−CI MFL model at a range of final P_{lac} induction levels. The panels on the left show results for simulations starting from a low O/N; those on the right for a high O/N. In the lower panels, the density of trajectories (number of simulated cells) producing given *pR* activities at final P_{lac} induction levels are displayed as a heat map. Deterministic trajectories are overlaid on each heatmap; the arrow indicates which of the hysteresis curves was simulated in the stochastic model. The upper panels show histograms for two sample conditions as indicated by the vertical lines in the lower panels; insets show the same distributions on a logarithmic scale to facilitate comparisons with the flow cytometry results. Parameters for the models are as listed in the 'Simple fit' column of Table 6.2, with the burst sizes of CI, Tum and GFP defined to be $\mathrm{b}_C = 300$ dimers, $\mathrm{b}_T = 3000$ monomers and $\mathrm{b}_Z = 20$ arbitrary units (a.u.) respectively.

P_{lac} induction levels that are outside the region of bistability predicted by the deterministic model. However, in both cases (i.e., in both low and high O/N simulations) bimodal distributions are observed. When plotted on a logarithmic axis (see insets), these bimodal histograms are more comparable with those obtained by flow cytometry, and reveal that even without instrumental noise, the low intensity and high intensity sub-populations still overlap.

The complete sets of low and high O/N simulated distributions are summarised using heat maps in the lower panels of Figure 6.5. The example histograms in the upper panels make up a single slice in each heatmap as indicated. These also mark the regions of the hysteresis curve where bimodal populations are found. For reference, both low and high O/N deterministic curves are plotted overlaid on each heatmap. There are broad regions of bimodality around each turning point of the deterministic curve, reflecting how stochastic switching occurs more readily near the deterministic bifurcation boundaries. The region of bimodality for the low O/N curve is much broader than that for the high O/N curve. This is consistent with the experimental results, though the bimodal low O/N distributions seen experimentally seemed to cover a broader range of induction levels still.

In order to investigate how switching from one stable state to the other occurs, the behaviour of individual stochastic trajectories over time was considered. Five sample time-course simulations each of low O/N and high O/N hysteresis trajectories are shown in Figure 6.6. The hysteresis conditions (i.e., the variations in P_{lac}) match those of the distributions shown in Figure 6.4, since they are located within the region of deterministic bistability. This means that reference lines showing the locations of the stable steady-states and also the unstable steady-state can be put in each plot. Trajectories ending near the low intensity stable state and others ending near the high intensity stable state can both be found. Importantly, in both the low O/N and high O/N simulations, a number of trajectories deviate significantly from the deterministic trajectory (thin black lines).

Switching from the high to the low intensity stable state, Figure 6.6(a), primarily relies on bursty production in Tum. With such a large burst size for Tum production ($b_T = 3000$ monomers), the time between bursts of Tum can be quite long even when production rates are high. If these times are long enough that the total Tum concentration drops below the unstable steady state, the system is able to switch to the lower stable state. Without this large Tum burst size, much less switching occurs. This can be seen by comparing the $b_T = 300$ and $b_T = 3000$ density curves in the upper panel of Figure 6.4(b). In contrast, switching from the low to the high intensity stable state, Figure 6.6(b), primarily relies on bursty production in CI. Even if bursty Tum production takes total Tum levels over the unstable state, there need to be sufficient delays in CI production such that the concentration of free CI (C) drops below its unstable state. Larger Tum burst sizes make no difference in this case as can be seen in the lower panel of Figure 6.4(b).

The averages of 5000 stochastic trajectories are also shown in Figure 6.6. These species

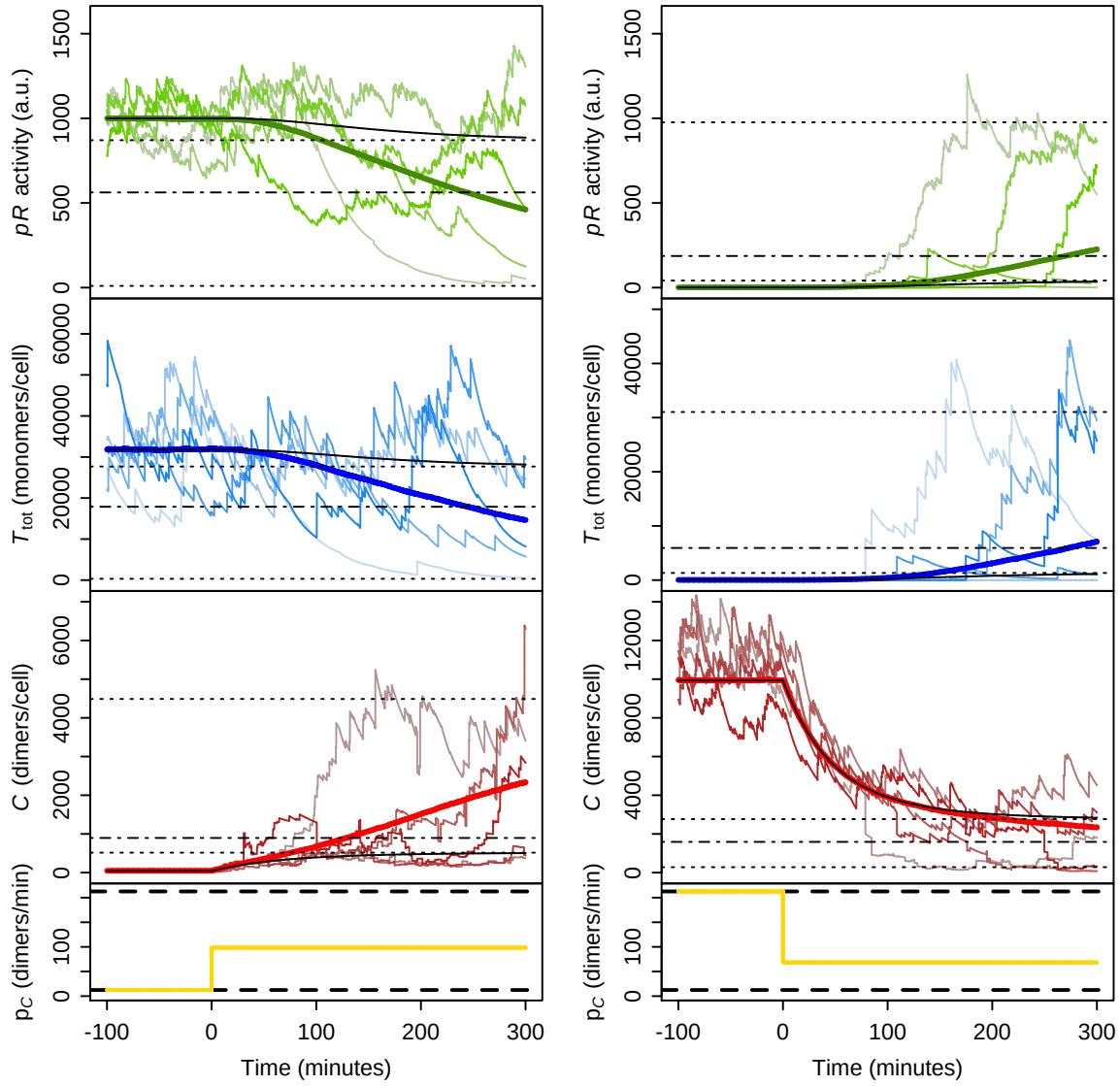


Figure 6.6: The timing of stochastic switching in the hybrid stochastic/deterministic model of the Tum−CI MFL. Two conditions are tested: the left panels show simulations of the transition from a low O/N culture to an intermediate CI production rate near the right bifurcation boundary; the right panels show the transition from a high O/N culture to an intermediate CI production rate near the left bifurcation boundary. The variation over time of P_{lac} promoter activity (p_C), free CI concentration (C), total Tum concentration (T_{tot}) and *pR* activity are shown. Five stochastic trajectories distinguished by shading are shown in each case; thick solid lines show the average of 5000 stochastic trajectories; thin black lines show the corresponding deterministic trajectory. In the plot of P_{lac}, thick black dashed lines indicate the activities ascribed to low and high O/N cultures. In the other plots, horizontal dotted lines indicate the stable steady-state values of that variable at the final level of induction, whilst the dot-dash lines indicate the unstable steady-state value. Parameters for the model are as listed in the 'Simple fit' column of Table 6.2, with the burst sizes of CI, Tum and GFP defined to be $\mathrm{b}_C = 300$ dimers, $\mathrm{b}_T = 3000$ monomers and $\mathrm{b}_Z = 20$ arbitrary units (a.u.) respectively.

averages give an indication of the proportion of trajectories that have switched to the alternative stable state. Interestingly, whilst the deterministic trajectories are seen to approach their equilibrium points, the average stochastic trajectories show no sign of slowing to some equilibrium point. In other words, stochastic equilibration between the two stable states occurs much more slowly than deterministic equilibration. This may partly explain why the high intensity sub-populations in the low O/N flow cytometry measurements appeared to equilibrate so slowly.

The simulated results can also be examined for sub-populations, which would reveal if slow equilibration of the high intensity states is also a feature of the model. Furthermore, such an analysis could be used to validate the treatment of the sub-populations as semi-deterministic descriptors of the data, as has been assumed in Section 5.3 for the steady-state model and in Section 6.3.1 for the dynamic deterministic model. Recall that in analysing the flow cytometry measurements, the Tum$^-$ control distributions were used to formulate an empirical description of noise as a function of median fluorescence (i.e., the constrained skew-*t* distributions). These constrained skew-*t* distributions were remarkably robust emulators of the fluorescence probability distribution, providing a simple way to approximate multiple underlying sources of noise, including instrumental noise, extrinsic sources of cellular noise, and, of course, the noise intrinsic to the gene network itself. The empirical description meant that complex models of noise propagation could be avoided when fitting the bimodal populations of the MFL strains. By similarly fitting the simulated stochastic data, a direct comparison of the fitted components can be made with the equivalent deterministic model to discern the effectiveness of such an empirical analysis.

Hysteresis was simulated for three alternative equilibration times, and the simulated fluorescence distributions fitted with mixture models of constrained skew-*t* distributions. The means of the fitted mixture model components are shown in Figure 6.7 with deterministic curves overlaid for comparison. For this analysis, minor amendments were made to the protocols that are briefly summarised below; technical details can be found in Appendix D. In order to reproduce experimental conditions more realistically, simulated cellular autofluorescence was added to the model data before mixture modelling. Autofluorescence predominates when reporter levels are low, so an ER-MC Tum$^-$ control flow cytometry distribution at maximum induction levels was used to estimate the magnitude of the background fluorescence. The normalised flow cytometry data in the untransformed scale was well fit by a normal distribution, so the mean and SD of that data were scaled to equivalent model units to define a theoretical distribution of background noise. Autofluorescence was presumed to be an additive source of noise, so simulated *pR* activities were offset by a random number drawn from the background noise distribution. The estimated burst size for GFP production was also revised to ensure that simulated *pR* activities were of the same order as the autofluorescence. After correcting for background

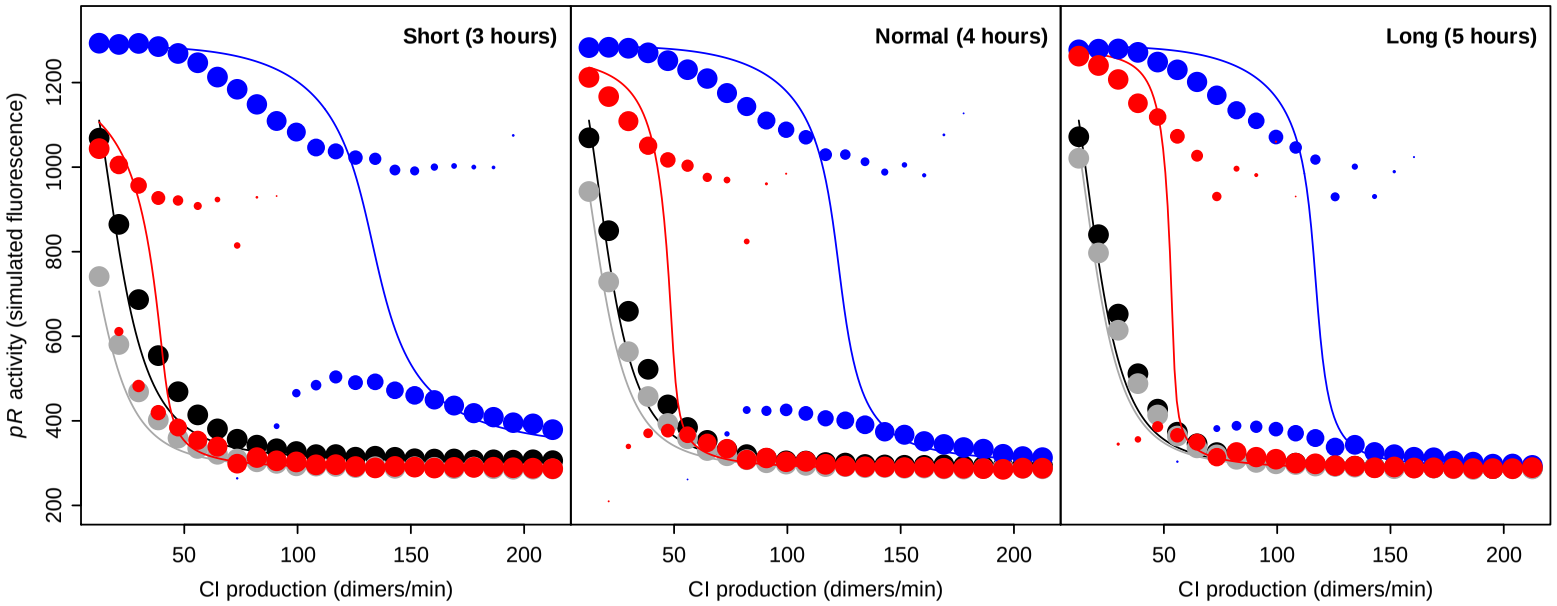


Figure 6.7: Sub-populations fitted to the simulated Tum−CI MFL data validate the semi-deterministic interpretation. Simulations of hysteresis for three different equilibration times were performed for both Tum$^-$ control and Tum$^+$ MFL models, and the entire set analysed as per the constrained skew-*t* mixture modelling described in Section 5.2. To better match the experimental results, random levels of background fluorescence, simulated from a normal distribution with mean 286 and SD 214, were added to the final *pR* activities, and the Logicle transform was applied to all distributions prior to mixture modelling. Populations were designated as unimodal or bimodal according to the log-likelihood ratio. Means for the component skew-*t* distributions were calculated in the untransformed scale, and are shown plotted versus the final levels of P_{lac} induction. The area of each point is proportional to population size. Deterministic curves are overlaid for reference. Parameters for all models are as listed in the 'Simple fit' column of Table 6.2, with burst sizes of CI, Tum and GFP in the stochastic models defined to be $\mathrm{b}_C = 300$ dimers, $\mathrm{b}_T = 3000$ monomers and $\mathrm{b}_Z = 100$ arbitrary units (a.u.) respectively.

fluorescence, the variance and mean fluorescence of an uninduced ER-MC Tum$^-$ control strain were calculated, and a suitable GFP burst size calculated using Equation (6.6). Simulated fluorescence distributions were generated for both Tum$^-$ control and Tum$^+$ MFL models and a Logicle transformation applied for consistency with the experimental analysis. The Tum$^-$ control simulations were used to calibrate constrained skew-*t* distributions for mixture modelling of the Tum$^+$ MFL simulations in the manner previously described in Section 5.2.

Though not a perfect match, the fitted sub-populations shown in Figure 6.7 do reproduce many features of the deterministic model. Clearly there is substantial ambiguity in the location of the deterministic turning points, but elsewhere many sub-populations are observed to lie directly on the deterministic curves, even equilibrating at the same rate. The most noticeable deviation is for the high intensity sub-populations of the low O/N. This deviation is likely to arise through deficiencies in both the mixture model and the deterministic model. The mixture models assume that no population of cells are in transition between the two stable states, whereas the deterministic model does not account for the distorted propagation of population means through nonlinear reaction pathways (i.e., the Tum$-$CI and CI$-$*pR* equilibria). The latter effect can be observed more directly for the simulated control distributions, where the ultrasensitive CI$-$*pR* equilibrium causes the stochastic means to diverge from the equivalent deterministic curves. In any case, the sub-population means were useful guides for calibrating the deterministic parameters to the data: the simulated results do compare relatively well to the exerimental sub-populations shown in Figure 5.11. Indeed, the slow rate of loss of high intensity populations at high induction levels is also apparent in the simulated results.

6.5 Improving and interpreting the stochastic model

The extent to which parameter variation could be exploited to obtain a better match between model and experiment is an open question. Statistical inference using the present stochastic model would take considerable compute time, since many thousands of simulated trajectories are required to obtain well-formed distributions. Future work in this area could focus on reformulating the hybrid stochastic/deterministic model as a CLE (continuous stochastic model). Solutions to the CLE might then be formulated as mixtures of Gaussian sub-populations to mimic the mixture model analysis. The CLE is an example of a Fokker-Planck equation, and mixture model solutions to that equation have been successfully calculated in other fields [Terejanu et al., 2008; Giza et al., 2009]. To the author's knowledge, such approaches have not yet been applied in the study of stochastic gene networks. Bayesian estimation of parameters using such solutions to the CLE should prove much more accurate than the deterministic methods of inference presented here and would also enable optimisation of stochastic parameters.

Parameter variation alone is unlikely to provide a satisfactory description of the experimental results. To obtain the degree of stochastic switching seen in this section, an unrealistic burst size for Tum production was required. The largest burst sizes seen in *E. coli* tend to range in the hundreds of proteins [Taniguchi et al., 2010], and at such sizes mRNA half-lives would be so long that the assumption of instaneous bursts of protein would also be invalid. Indeed, the difference in bursting dynamics between Tum and GFP in Figure 6.6 is enough cause for concern, since both proteins are produced from the same promoter. In fact, the rate of firing of the Tum promoter is expected to be *more* frequent than that of GFP, since Tum is located on a multi-copy plasmid. Instead, the requirement for such a large burst size suggests that this parameter is acting as a proxy for some other stochastic reaction that operates on a timescale slower than mRNA production (translation bursts).

Likely candidates for such slow stochastic reactions are the replication and partitioning by cell division of the Tum plasmid. Small changes in plasmid copy number can lead to large relative changes in gene output [Mileyko et al., 2008], and changes in plasmid copy number have even been observed to produce qualitative changes in gene circuit behaviour [Anand et al., 2011]. With only 10–12 copies per genome, small changes in the copy number of the pMTS-*pR-tum*$^+$ MFL plasmid could lead to large relative changes in promoter output, which would then be amplified by translation. To investigate this possibility, the stochastic model would need to be extended to simulate both the replication of promoter elements and binomial partitioning of the plasmid components. Stable plasmid propagation requires regulated replication, and a simple negative feedback mechanism is sufficient to account for this in stochastic models [Paulsson and Ehrenberg, 2001]. Such extensions to the model will be investigated in future work.

7

The MFL now and going forwards

The MFL is over-represented in natural networks [Yeger-Lotem et al., 2004], and is a paradigm for regulatory motifs that cross between transcriptional and protein-protein interaction networks. In this thesis, a synthetic MFL that displays bistability was constructed in *E. coli* from the Tum and CI proteins of phage 186. These proteins are well-studied [Lamont et al., 1989; Shearwin and Egan, 1996; Shearwin et al., 1998; Dodd and Egan, 2002; Pinkett et al., 2006] and show few interactions with the host *E. coli* network, so presented an ideal choice for testing the functions of the MFL network in isolation.

A deterministic model of the Tum−CI MFL was developed in Chapter 2 using the information available from the literature on the regulatory capacities of Tum and CI. This model showed that a MFL constructed from these proteins was theoretically capable of producing a network that would display bistability. This model also inspired a hysteresis assay that could be implemented experimentally to efficiently balance parameters and locate a bistable configuration. In contrast with the generalised deterministic model of the bistable MFL developed by François and Hakim [2005], the deterministic model developed in Chapter 2 was a more accurate representative of the Tum−CI MFL design. Furthermore, it highlighted a hysteretic variable not considered by François and Hakim: whereas they used the production rate of the transcriptional regulator as a normalisation factor, here that production rate is varied to achieve hysteresis. This hysteresis variable was easily varied by way of a LacI induction system controlling expression of the CI repressor. It also motivated a strain lacking the *tum* gene as a useful control for comparing the hysteresis of the bistable system with a monostable one.

The use of a deterministic model to aid the design of a synthetic circuit is a common pattern in synthetic biology. The first synthetic circuits — the oscillator of Elowitz and Leibler [2000] and the toggle switch of Gardner et al. [2000] — both made use of simple models of gene networks to guide the construction process. Though deterministic models are not as accurate as a complete stochastic model of a network, their simplicity and analytical tractability makes them integral conceptual tools. The deterministic model of the Tum−CI MFL was used throughout this thesis as a standard against which to measure the performance of the MFL. In this way, it guided the experiments towards a more complete understanding of the synthetic circuits. Firstly, the deterministic model helped to identify

a set of core parameters that could set bounds on how the results could be interpreted. Experimental measurements of these parameters were reported in Chapter 3. Then, based on the performance of the controls, it was found that longer equilibration times were needed, and the hysteresis protocol was modified to accommodate this as reported in Chapter 4. Direct comparison of the deterministic model with the results of whole-population hysteresis measured by LacZ assay prompted measurements of the distribution of reporter activities in single cells. These distributions were measured by flow cytometry in Chapter 5, and a method was devised for splitting the observed bimodal populations into effective deterministic components. These components compared favourably with the deterministic model, underscoring how such models can still be relevant in a stochastic system. Finally, by comparing the deterministic model with a hybrid stochastic/deterministic model of the Tum−CI MFL developed in Chapter 6, the limits of the deterministic model could be assessed. Though it failed to capture minor deviations due to noise propagation through a nonlinear network, the deterministic model matched the general trends of the data surprisingly well.

Another common theme found throughout the thesis was the normalisation of experimental data in order to better capture the underlying behaviour of the MFL. Prime examples of this included normalisation of the ultrasensitivity present in the induction system, morphology normalisation of the flow cytometry data sets and the constrained skew-*t* mixture modelling. By measuring the ultrasensitivity present in the relief of LacI repression of P_{lac} by IPTG in Section 3.3, all future measurements of hysteresis could be normalised to a linear scale in the hysteresis variable. Such a normalisation method was previously used to more accurately measure downstream ultrasensitivity of an activator [Palmer et al., 2009], and in this thesis it made for better alignment of the data with both deterministic and stochastic models. A wealth of data is captured in processing cell cultures by flow cytometry, and this makes it possible to correct for noise due to variation in cell morphology [Knijnenburg et al., 2011]. By applying morphology normalisation to the Tum−CI MFL data sets, the distribution of fluorescence activities became distributions for an 'average' cell. This improved resolution of the bimodal samples, and facilitated the interpretation of these distributions in terms of deterministic components, since deterministic models reduce complexity by assuming a constant cell volume. The constrained skew-*t* mixture model developed in Section 5.2 then provided the means for precise determination of sub-population locations and proportions. By building on the *pR* distribution template provided by the *tum*$^-$ control strains, the constrained skew-*t* distributions were apt proxies for the unimodal distributions that would be expected of a deterministic trajectory. This ultimately enabled direct comparison of an obfuscated mixed distribution with the simple deterministic model.

Regulatory networks bearing similarities with the bistable MFL have been studied in recent years. The regulatory network for galactose metabolism in *S. cerevisiae* contains a

MFL with overall negative feedback involving the Gal4p activator which gets inactivated by the Gal80p protein by a protein-protein interaction [Smidtas et al., 2006]. This would normally imply a MFL in the oscillatory regime, but two additional positive feedback loops make the overall network behave as a bistable system. These positive feedback loops involve the sequestration of Gal80p by Gal3p and Gal1p, which makes each of these three-component feedback loops similar to the MFL motif, but with an additional intervening protein-protein interaction. The bistable configuration of these three-component circuits has been shown to produce hysteresis as a function of galactose concentration [Venturelli et al., 2012], and, like for the Tum−CI MFL, bimodal populations were observed throughout the bistable region of the hysteresis curve.

A simple synthetic switch involving the sequestration of a sigma factor by its anti-sigma factor has also been constructed in *E. coli* [Chen and Arkin, 2012]. In this case, the sigma factor itself forms an overall positive feedback loop by activating its own production; the anti-sigma factor acts to adjust the strength of positive feedback by sequestration of the sigma factor. Chen and Arkin observe hysteresis in this synthetic bistable switch, but again a bimodal population structure is found in the bistable region. Importantly, this synthetic circuit is not an example of the MFL, in which the transcriptional regulator acts to control expression of the protein interaction partner. Instead the sigma/anti-sigma circuit constitutes an overall transcription-based positive feedback loop that is modulated by sequestration.

These examples and the Tum−CI MFL all produce bimodal populations in cycling the hysteresis loop. The Tum−CI MFL, in particular, has very closely overlapping populations that are difficult to resolve with basic mixture modelling. Is such noisy switching a feature of these sequestration-based circuits, or is it possible to construct a robust bistable circuit with the MFL motif? To begin addressing this question, some modifications to the design of the Tum−CI MFL were proposed. As found previously in Section 6.3.2, noise in CI production plays a prominent role in stochastic switching near the bifurcation boundaries, and even a moderate reduction of noise in this parameter could have a significant impact on switch robustness. On this basis, an improved P_{lac}-*c*I induction module was designed using the Lac repressor looping constructs described by Priest et al. [2014]. By maintaining both the source of Lac repressor and also the P_{lac} induction module in the chromosome, the copy-number of these elements should be better linked to cell growth, rather than to the stochastic copy-number variation typical of plasmids [Paulsson and Ehrenberg, 2001]. Furthermore, in order to enable estimation of the intrinsic stochasticity of the integrated P_{lac} module, a P_{lac}-*gfp* reporter construct was designed. The other *pR* promoter reporter modules were also redesigned to make use of an improved integration chassis, the OSIP (One-Step Integration Plasmid) system [St-Pierre et al., 2013], and to replace the folding reporter GFP with the brighter superfolder GFP [Pédelacq et al., 2006] to increase the sensitivity of the fluorescence assays. I have cloned and integrated most of these new con-

structs into an improved induction and reporter strain. However, a complete MFL strain has not yet been constructed, so the details of these strains and the pending assays will be reported elsewhere.

These proposed improvements to the Tum−CI MFL still do not address what could be the most important source of noise: copy-number variation of the *pR-tum* plasmid. Integration of this module would limit such variation, but recall that an integrated *pR-tum* module is not strong enough to produce discernable hysteresis [Pocock, 2007]. Even if the strength of the P_{lac} promoter could be lowered to the point where CI and Tum production were balanced, it is likely that the lower rate of promoter firing would give rise to larger relative fluctuations that would increase stochastic switching in any case. Solutions may instead include either increasing the degradation rate of CI by appending a degradation tag to the C-terminal end of the protein [McGinness et al., 2006], or increasing the degradation rate of the mRNA using small RNAs [Shimoni et al., 2007]. In these cases, high rates of promoter firing could be maintained for reduced steady-state levels of CI repressor. This may allow integration of the *pR-tum* module without concomitantly increasing noise in CI expression.

There remains much to learn about the MFL. Though robust bistability from this motif has not yet been observed, this may not be its primary function. Indeed, having a bimodal phenotype in a bacterial population may improve the chances of survival, as exemplified by bacterial persister cells [Balaban et al., 2004]. In *Drosophila*, random segregation into bimodal populations is responsible for the differentiation of photoreceptor cells into two different types [Mikeladze-Dvali et al., 2005]. Though the choice is random for any given cell, the relative fraction of each cell type is well defined. Alternatively, the MFL motif may be important in some as-yet-unidentified signal processing role. Such features are unlikely to be visible from the relatively static picture of the MFL obtained here by LacZ assay or flow cytometry. Instead, full time-course measurements of single cells and their responses to transient signals would be required. A microfluidics platform in combination with fluorescence-based reporting, like the systems used to monitor synthetic oscillator circuits [Stricker et al., 2008; Prindle et al., 2012], could provide such experimental flexibility, opening the way to studying dynamic signal propagation and triggering of the switch. Given these open questions, the construction of a functional synthetic Tum−CI MFL, as reported in this thesis, stands as an important first step towards understanding the importance of the MFL motif.

8

Materials and Methods

8.1 Reagents

All reagents used in this thesis were of analytical grade or the highest purity available. Standard chemicals are listed in Table 8.1. Standard buffers and media prepared from these chemicals are listed in Table 8.2.

Table 8.1: Standard chemicals used in this thesis.

Name	Abbrev.	Company	Notes
General salts, acids and bases			
Calcium chloride	$CaCl_2$	Sigma (Sigma Chemical Co.)	
Di-sodium hydrogen orthophosphate	Na_2HPO_4	B.D.H. (B.D.H. Labs, Australia)	
Hydrochloric acid	HCl	B.D.H.	
Magnesium Chloride	$MgCl_2$	Ajax	
Magnesium sulphate	$MgSO_4$	Ajax	
Milli-Q water	H_2O		Water purified using Millipore Corporation filters
Potassium chloride	KCl	B.D.H.	
Monopotassium phosphate	KH_2PO_4	Sigma	
Sodium acetate		B.D.H.	
Sodium chloride	NaCl	B.D.H.	
Sodium hydroxide	NaOH	Ajax	
Tris acetate		B.D.H.	
Biochemicals			

Continued on next page

Table 8.1: *Continued from previous page*

Name	Abbrev.	Company	Notes
2-Log DNA molecular markers	2-log ladder	NEB (New England Biolabs)	
Agarose		Sigma	
Bacto-agar		Difco (Difco Laboratories Inc.)	
Bacto-tryptone		Difco	
Bovine Serum Albumin	BSA		
Casamino acids		Difco	
Deoxyribonucleoside triphosphates	dNTPs	NEB	
Glucose		Ajax	
Glycerol		B.D.H.	
Yeast extract		Difco	
Antibiotics			
Ampicillin	Amp	Sigma	Stock solutions (100 mg/mL in H_2O) were Millipore filtered and stored at $-20°C$.
Chloramphenicol	Chlor	Sigma	Stock solutions (30 mg/mL in ethanol) were stored at $-20°C$.
Kanamycin	Kan	Sigma	Stock solutions (50 mg/mL in H_2O) were Millipore filtered and stored at $-20°C$.
Spectinomycin	Spec	Sigma	Stock solutions (50 mg/mL in H_2O) were Millipore filtered and stored at $-20°C$.
Tetracyclin	Tet	Upjohn Pty. Ltd.	Stock solutions (20 mg/mL in ethanol) were stored at $-20°C$.
Other chemicals and solvents			
B-PER® Reagent		Thermo Scientific (Thermo Fisher Scientific Inc.)	
Ethanol		Crown Scientific	RNase-free

Continued on next page

Table 8.1: *Continued from previous page*

Name	Abbrev.	Company	Notes
Ethylenediaminetetraacetic acid (disodium salt)	EDTA	Sigma	
Isopropanol		May and Baker Ltd.	
GelRed		Biotium	
5-bromo-4-chloro-indolyl-β-D-galactopyranoside	X-gal	Sigma	Stock solutions (30 mg/mL in dimethyl formamide) were stored at −20°C.
Isopropyl-β-D-1-thiogalactopyranoside	IPTG	Sigma	Stock solutions (100 mM in H_2O) were Millipore filtered and stored at −20°C.
2-mercaptoethanol	β-ME	Sigma	Stored at −20°C.
ortho-nitrophenyl-β-galactosidase	ONPG	Diagnostic Chemicals Ltd.	Stored as a powder at −20°C. Solutions always made fresh.
polymixin B sulphate	PMB	Sigma	Stock solutions (20 mg/mL in H_2O) were stored at −20°C.
Sodium dodecyl sulphate	SDS	Sigma	
Enzymes			
Antarctic Phosphatase		NEB	
Benzonase		Novagen	Stock concentration is 25 U/μL.
BigDye Version 3.1 Ready Mix		Life Technologies	
Lysozyme		Sigma	
T4 DNA ligase		NEB	
Taq DNA polymerase		NEB	
Phusion High-Fidelity DNA polymerase		Finnzymes	
Restriction endonucleases		NEB	

Table 8.2: Standard buffers and growth media used in this thesis. Buffers were prepared in H_2O unless otherwise specified.

Name	Abbrev.	Formulation
BigDye Dilution buffer		200 mM Tris-HCl pH 9.0, 5 mM $MgCl_2$.

Continued on next page

Table 8.2: *Continued from previous page*

Name	Abbrev.	Formulation
10× Glycerol Loading Buffer	GLB	50% (v/v) glycerol, 0.4% (w/v) Bromophenol Blue, 0.2% (w/v) Xylene Cyanol, 10 mM EDTA.
L agar		1.5% Bacto-agar in LB. Autoclaved.
Lennox Luria broth	LB	1% bacto-tryptone, 1% NaCl, 0.5% yeast extract, pH 7.0. Autoclaved.
10× M9 salts		128 g/L Na_2HPO_4, 30 g/L KH_2PO_4, 10 g/L NH_4Cl, 5 g/L NaCl. Autoclaved.
M9 minimal media		1× M9 salts, 0.1 mM $CaCl_2$, 1 mM $MgSO_4$, 0.2% (w/v) glucose or glycerol, and optionally 0.2% (w/v) casamino acids and/or 1 μg/mL thiamine.
1× Phosphate Buffered Saline	PBS	137 mM NaCl, 2.7 mM KCl, 10 mM Na_2HPO_4, 1.8 mM KH_2PO_4, pH 7.4. Filtered to remove debris.
SOC media		20 g/L tryptone, 5 g/L yeast extract, 10 mM NaCl, 2.5 mM KCl, 10 mM $MgCl_2$, 10 mM $MgSO_4$. Autoclaved.
10× TAE		0.4 M Tris-acetate, 0.2 M sodium acetate, 10 mM EDTA, pH 8.2
TSS		10% (w/v) polyethylene glycol (M.W. 8000), 5% (v/v) dimethyl sulfoxide, 50 mM $MgCl_2$ made up in LB at pH 6.5; either autoclaved or filter sterilised using a 0.2 μm filter. Stored at 4°C in the short term, or −20°C for the long term.
TZ8		100 mM Tris-HCl pH 8.0, 10 mM KCl, 1 mM $MgSO_4$ in H_2O.
TZ8+		50 μg/mL polymixin B (PMB) and 0.27% (v/v) 2-mercaptoethanol (β-ME) made up in TZ8. Always made fresh.

8.2 General cloning methods

8.2.1 Growth of bacteria

Unless otherwise specified, bacterial cultures and colonies were grown at 37°C. Strains containing the CRIM helper plasmids were grown at 30°C due to the temperature-sensitive replication origin.

Liquid cultures were routinely prepared in sterile Lennox Luria broth (LB) with added antibiotics unless otherwise specified. Single colonies were obtained by streaking or spread-

Table 8.3: Concentrations of antibiotics used in this thesis. Concentrations are specified in μg/mL. The concentration used depended on plasmid copy number; for integration of the CRIM plasmids alternative concentrations were also required. High-copy plasmids were those with either the ColE1, R6Kγ, oriR101 or p15a replication origins. Low-copy plasmids were those with the mini-F, SC101 or SC101* replication origins.

Antibiotic	High copy	Low copy	Integration
Ampicillin	100	30	–
Chloramphenicol	30	15	4
Kanamycin	50	40	20
Spectinomycin	50	25	10
Tetracyclin	20	4	2

ing onto L-plates. L-plates were prepared by adding the required antibiotics to molten 1.5% L agar at 50°C, pouring into petri dishes, and drying O/N at 37°C. For plasmid maintenance and integration of the CRIM plasmids, antibiotics were added to the required growth medium at the concentrations listed in Table 8.3. After successful integration of CRIM plasmids, strains were routinely grown without selection for the integrated plasmid's antibiotic resistance.

Stationary phase cultures were prepared by innoculating liquid growth medium with a freshly-streaked colony and incubating overnight (O/N) with shaking. Log phase cultures were prepared from stationary phase cultures by diluting 1/200 in liquid growth medium and incubating with shaking until the culture reached an optical density (absorbance at 600 nm) of 0.6 (OD_{600} 0.6).

Optical densities were routinely measured using a Cary 3-Bio UV-Vis Spectrophotometer (cultures diluted 1/10 in water before measurement). Where more accuracy was required, 100 μL samples of the culture were pipetted into the wells of a flat-bottomed 96-well plate and the absorbance at 620 nm (A_{620}) was measured using a Multiskan Ascent plate reader (Labsystems). The A_{620}s were converted to OD_{600}s using an empirically determined relationship [Dodd et al., 2001].

8.2.2 Storage of bacterial strains

For long-term storage of bacterial strains, 500 μL of an O/N culture was mixed with 500 μL of 80% glycerol and stored at −80°C in sterile 1.5 mL screw-capped tubes. Strains were retrieved by streaking directly from the frozen stocks onto L-plates with relevant antibiotics for growth O/N.

8.2.3 Preparation and purification of DNA

Plasmid DNA was usually obtained using a small-scale preparation. Plasmid DNA was extracted and purified from pelleted 4 mL O/N cultures (4000 rpm, 10′) using the QIAprep

Spin Miniprep Kit (Qiagen), following the protocol as specified. Plasmids were eluted in 50 μL of the provided buffer (10 mM Tris-HCL, pH 8.5) and stored at −20°C.

For cloning into plasmids with the mini-F origin (i.e., those based on pZC320), large-scale preparation of plasmids was performed using the NucleoBond Xtra Midi Plus kit (Macherey-Nagel). The kit's Low-copy Plasmid Purification protocol was followed for O/N cultures of 400 mL. Plasmids were concentrated with the NucleoBond Finalizer, eluted in 300 μL of 5 mM Tris-HCl, pH 8.5, and stored at −20°C.

8.2.4 Polymerase Chain Reactions

For amplification of DNA for cloning, polymerase chain reactions (PCRs) with the Phusion High-Fidelity DNA polymerase (Finnzymes) were used. Reactions were made up to 20 μL in H_2O with 0.5–20 ng of template DNA, 0.2 mM of each dNTP, 1× HF Phusion Buffer, 40 ng of each primer and 0.4 U of polymerase. Temperature cycling was performed using a DNAEngine Thermal Cycler (BioRad) with the cycling protocol set to match the manufacturer's recommendations for the Phusion polymerase.

For cloning work, products from PCR reactions were purified using either the MoBio UltraClean PCR Cleanup Kit, typically eluting in 50 μL 10 mM Tris-HCl, or the DNA Clean & Concentrator-5 kit (Zymo Research), typically eluting products in 10 μL H_2O.

For amplification of DNA to prepare sequencing templates or to screen for clones, PCRs with the Taq DNA polymerase (NEB) were used. Templates for these reactions included both genomic DNA and plasmids, which were included in the reaction either by pipetting from a plasmid prep (to obtain 0.5–20 ng total in the reaction), picking from colonies on L-plates using a sterile wire and stirring into the reaction mix, or by adding 2 μL of a cell suspension prepared by picking colonies on L-plates using yellow tips and resuspending in 100 μL of H_2O. Reactions were prepared in thin-walled 0.2 mL PCR tubes and made up to 10 μL (screening) or 20 μL (sequencing) in H_2O with template, 0.2 mM of each dNTP, 1× ThermoPol buffer (NEB), 2 ng/μL of each primer, and 0.05 U/μL of Taq polymerase. Temperature cycling was performed using a Rapid Cycler from Idaho Technology. For amplification of products with lengths smaller than 1 k.b., the cycling protocol was set to hold 98°C for 10 seconds, then 30 cycles of (98°C for 10 seconds, 46°C for 10 seconds and 74°C for 15 seconds). For amplification of products with lengths greater than 1 k.b., the extension time at 74°C was increased to 1 minute.

For screening of CRIM plasmid integrants, the protocol for screening colonies was followed, but the four primers were instead combined to be equimolar with final concentrations of 0.5 μM each.

8.2.5 Analysis of DNA

DNA preparations (including plasmid and PCR) were routinely analysed by agarose gel electrophoresis. Gels were prepared by dissolving 1–2% agarose in boiling 1×TAE buffer

and casting this into horizontal minigels. Gels were run at 110 V in a Mini Sub Cell GT (BioRad) electrophoresis device using 1×TAE as running buffer. Typically, 2–4 μL of a mixture of DNA sample and GLB (1×) were loaded into each well; at least one lane contained 500 ng of 2-log DNA molecular markers (NEB) made up in GLB.

DNA gels were visualised by staining with GelRed (Biotium) according to the manufacturers instructions, and photographing under short wavelength UV light. The molecular markers were used to estimate both lengths and concentrations (by comparison of band intensities) of the DNA samples.

8.2.6 DNA recombination work

Restriction endonucleases were obtained from New England Biolabs (NEB) and DNA digests were performed as per the manufacturer's recommendations in 50 μL reaction volumes with incubation times ranging from 1 hour to overnight. Digestion was confirmed using agarose gel electrophoresis.

To avoid self religation, the ends of linearised plasmid backbones were routinely dephosphorylated using Antarctic Phosphatase (NEB). 5 U of phosphatase and sufficient Antarctic Phosphatase buffer were added to restriction digests with incubation at 37°C for 30 minutes. The enzyme was heat inactivated by incubating at 65°C for 5 minutes.

Prior to ligation, linearised DNA was usually gel extracted to isolate just the intended fragment. Digestion reactions were separated by agarose gel electrophoresis using 1% agarose minigels cast with larger wells. Gels were stained with 50 mL of 1× SYBR Safe DNA gel stain (Life Technologies) made up in TAE for 30 minutes with gentle shaking. Excision of the desired bands was assisted by visualisation using a Safe Imager Blue-Light Transilluminator (Life Technologies). DNA was extracted from gel slices using either the QIAquick Gel Extraction Kit (Qiagen), eluting in 30 μL of 10 mM Tris-HCl, or the Zymoclean Gel DNA recovery Kit (Zymo Research), eluting in 10 μL of H_2O.

For ligation of DNA, 10 μL reactions were prepared using 200 U of T4 DNA ligase (NEB), 1 μL ligase buffer (10× stock), with the remaining volume split between insert and vector DNA solutions to achieve an approximate 3:1 ratio of insert to vector. Controls without the insert and without ligase were also prepared to determine the background. Ligation reactions were desalted by pipetting onto MF-Millipore Membrane Filter disks with a 0.025 μm pore size that were floating in H_2O. Reactions were removed after 15 minutes and pipetted into clean microcentrifuge tubes.

8.2.7 Competent cells

Electrocompetent transformation. For cloning work, electrocompetent transformation of desalted ligation reactions was used due to the higher transformation efficiency of this method.

Electrocompetent cells were prepared from a 500 mL log phase culture (OD_{600} 0.4–0.6). The culture was divided into four equal volumes, the cells pelleted (2500 $\times$ g, 10 minutes, 4°C) and the supernatant removed. Each pellet was resuspended in 125 mL of ice-cold H_2O before pelleting again (2500 $\times$ g, 10 minutes, 4°C). The supernatant was removed, and each pellet resuspended in 1 mL of ice-cold 10% glycerol and transferred to 1.5 mL microcentrifuge tubes. Cells were again pelleted (6000 $\times$ g, 3 minutes, 4°C) and the supernatant removed. Cells were consolidated into two portions by resuspending in 2 aliquots of 1 mL 10% glycerol. Cells were pelleted a final time (6000 $\times$ g, 3 minutes, 4°C), resuspended in ice-cold 10% glycerol, aliquoted into 250 µL portions and stored at −80°C.

For transformation, cells were thawed on ice and a 40 µL aliquot mixed with 2 µL of ligation mix using chilled pipette tips. This mix was transferred into a chilled electroporation cuvette (1 mm gap) and an electric pulse applied using the Ec1 setting (1.8 kV pulse) of a Bio-Rad Micropulser. 1 mL of SOC with 20 mM glucose was immediately added into the cuvette and used to pipette the cells into 1.5 mL microcentrifuge tubes. Cells were allowed to recover by incubating at 37°C for 1 hour in a rotating drum. Cells were pelleted (6000 $\times$ g) and 850 µL of the supernatant discarded. Cells were resuspended in the remaining supernatant, spread onto L-plates with appropriate antibiotics and left to grow overnight at 37°C.

TSS competent transformation. For transformation of plasmids, including integration of the CRIM plasmids, TSS competent cells were prepared [Chung et al., 1989]. A 10 mL early log phase culture (OD_{600} 0.3–0.4) was chilled on ice for 30 minutes, and cells then pelleted (4000 rpm, 10 minutes, 4°C, Eppendorf Centrifuge 5810R). Cells were resuspended in 1 mL of TSS buffer, aliquoted into microcentrifuge tubes and stored at −80°C.

For transformation, the competent cells were thawed on ice and 100 µL mixed with typically 4 µL of plasmid mini prep using chilled pipette tips. After 20 minutes on ice, 900 µL of LB with 20 mM glucose was added and cells allowed to recover by incubating at 37°C for 1 hour in a rotating drum. Cells were pelleted, resuspended in a smaller volume and spread onto L-plates as for the electrocompetent transformation.

8.2.8 Sequencing

Sequencing was routinely used to check for the integrity of recombinant DNA constructs that had been prepared using a PCR product. Sanger sequencing reactions were prepared from templates that had first been PCR amplified. Reactions were prepared with 5 µL of cleaned PCR template, 7 µL of BigDye Dilution buffer, 2 µL of the primer (from a 20 ng/µL stock) and 1 µL of BigDye Version 3.1 Ready Mix, made up to 20 µL with H_2O. Thermal cycling was applied using an Idaho Technology Rapid Cycler (98°C for 20 seconds, then 30 times (96°C for 10 seconds, 47°C for 20 seconds, 57°C for 1 minute)).

Reactions were then cleaned using isopropanol precipitation. Reactions were transferred to microcentrifuge tubes and 80 μL of 75% isopropanol added. This was vortexed and then left to precipitate for 15′ at room temperature. Samples were spun in a marked orientation at 13.2 krpm for 10 minutes at room temperature in an Eppendorf Centrifuge 5415R. The supernatant was removed, the pellet washed in 250 μL 75% isopropanol and the tube vortexed. Samples were spun in same orientation for another 5 minutes and the supernatant again removed. This was repeated for another 2 minutes and any residual supernatant removed. Samples were then dried on a heating block with lids open for 3′ at 75°C before delivering to the Institute of Medical and Veterinary Sciences, South Australia (IMVS) for sequencing. Sequencing chromatographs were visualised using the ApE plasmid editor (`http://biologylabs.utah.edu/jorgensen/wayned/ape/`).

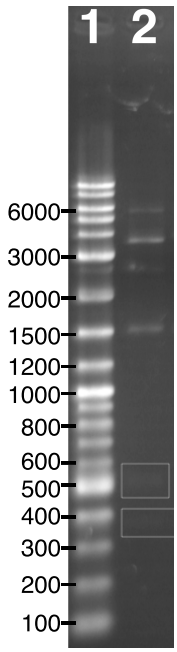
8.2.9 Changing resistance genes for the *pR-tum* plasmid

The *pR-tum* plasmids of the original Tum−CI MFL strain prepared by Pocock [2007] had a chloramphenicol resistance gene. This conflicted with the (single copy) chloramphenicol resistance of the integrated pIT-CH-*pR-gfp* construct. To avoid this conflict, the resistance gene on the pMCS-*pR-tum*$^+$ and pMCS*-*pR-tum*$^+$ constructs was changed to a gene conferring tetracycline resistance. Unfortunately, this step required a three fragment ligation reaction due to the presence of an extra SpeI site in the SC101 replication origin. This additional challenge meant that this cloning step required more care than usual, so it is described here in more detail.

The tetracycline resistance was derived from pZE55-λ *c*I, which was digested with AatII and SpeI and the tetracycline fragment gel extracted. The pMCS-*pR-tum*$^+$ and pMCS*-*pR-tum*$^+$ constructs were also digested with AatII and SpeI and combined with the tetracycline resistance fragment in the ligation reaction in roughly equimolar amounts. The most successful ligation reactions omitted gel extraction of the two fragments from the pMCS-*pR-tum*$^+$ and pMCS*-*pR-tum*$^+$ digests, that is, after heat inactivation of the restriction enzymes, the digests were used directly in the ligation reaction. Ligation reactions were transformed into electrocompetent E2878 pZC320-WR-*c*I, which has a source of CI protein to reduce the chance of promoter or gene mutations due to the strong *pR* promoter.

Since the secondary SpeI site was in the middle of the replication origin, it was unlikely that plasmids with a flipped SpeI-SpeI fragment could successfully propagate. Nonetheless, a number of diagnostic tests were run to ensure that the plasmids were correct. Mini preps of a selection of pMTS-*pR-tum*$^+$ and pMTS*-*pR-tum*$^+$ were prepared for diagnostic digests[1]. An NdeI/XhoI restriction digest reaction was chosen since it produced diagnos-

[1]NB: the same procedure was also trialled for changing the resistance genes of the original pMCS-*pR-tum*$^-$ and pMCS*-*pR-tum*$^-$ plasmids, so diagnostic digests of these plasmids were also performed. However, these clones were later discarded in favour of a more complete deletion of the *tum* gene from the completed pMTS-*pR-tum*$^+$ and pMTS*-*pR-tum*$^+$ plasmids as described in Section 8.3.3.



1. 500 ng 2-log ladder
2. 3 μL pMTS-*pR-tum*$^+$ C1 AatII/XhoI/NdeI digest

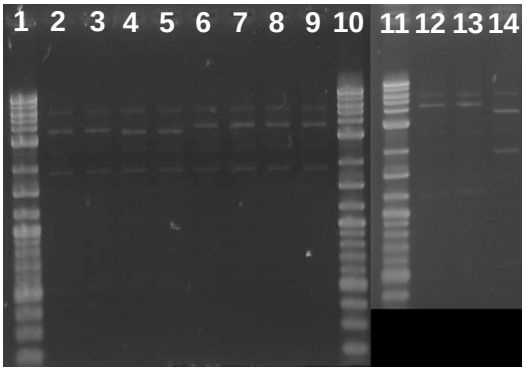
Figure 8.1: 1.5% agarose gel of diagnostic digest of pMTS-*pR-tum*$^+$ plasmid with AatII, XhoI and NdeI.

tic fragment lengths that tested how the fragments had ligated together. This digest was performed for all but one of the pMTS-*pR-tum*$^+$ clones, which was subject to digestion with AatII, XhoI & NdeI, which would produce the following fragments:

Ligation Variants	Expected Fragments
pMTS-*pR-tum*$^+$	3.6 k.b., 1.5 k.b., 0.5 k.b., 0.35 k.b.
pMTS-*pR-tum*$^+$ (reversed *tum* fragment)	3.9 k.b., 1.2 k.b., 0.5 k.b., 0.35 k.b.

A gel showing this digestion for Clone 1 (C1) from the pMTS-*pR-tum*$^+$ three fragment ligation reaction is shown in Figure 8.1.

For the clones digested only with NdeI and XhoI, the expected fragment sizes depending on how the ligation proceeded would be:



1. 500 ng 2-log ladder
2–5. 3 μL pMTS*-*pR-tum*$^+$ C1–4 NdeI/XhoI digests
6–9. 3 μL pMTS*-*pR-tum*$^-$ C1–4 NdeI/XhoI digests
10. 500 ng 2-log ladder
11. 500 ng 2-log ladder
12–14. 3 μL pMTS-*pR-tum*$^+$ C3–5 NdeI/XhoI digests

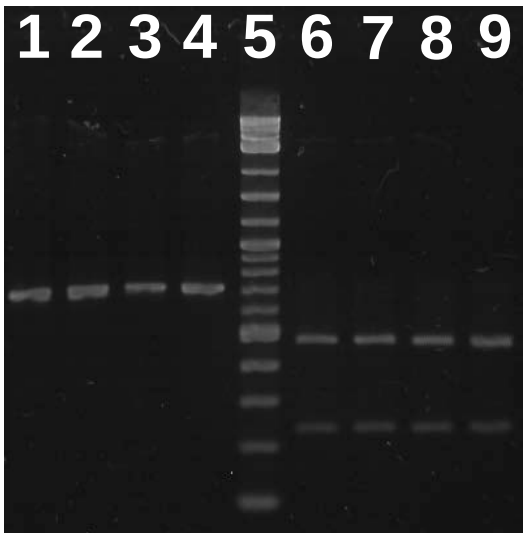
Figure 8.2: 1.5% agarose gel of diagnostic digest of *pR-tum* plasmids with NdeI/XhoI.

Ligation Variants	Expected Fragments
pMTS-*pR-tum*$^+$/ pMTS*-*pR-tum*$^+$	3.6 k.b., 1.9 k.b., 0.5 k.b.
pMTS-*pR-tum*$^+$/ pMTS*-*pR-tum*$^+$ (reversed *tum* fragment)	4.2 k.b., 1.2 k.b., 0.5 k.b.
pMCS-*pR-tum*$^+$/ pMCS*-*pR-tum*$^+$	3.6 k.b., 1.3 k.b., 0.5 k.b.
pMTS-*pR-tum*$^+$/ pMTS*-*pR-tum*$^+$ (double *tum* fragment)	3.6 k.b., 2.5 k.b., 1.9 k.b., 0.5 k.b.
pMTS (no *tum*/*tum*$^-$ fragment)	2.9 k.b.
pZC320-WR-*c*I	5.6 k.b., 2.5 k.b.

A gel showing these digestions for clones from the pMTS-*pR-tum*$^+$ and pMTS*-*pR-tum*$^+$ three fragment ligation reactions is shown in Figure 8.2. The gel revealed that pMTS-*pR-tum*$^+$ clones 3 and 4 had a reversed *tum* fragment. So the clones consistent with the correct order and orientation of the fragments were pMTS-*pR-tum*$^+$ clones 1&5, and pMTS*-*pR-tum*$^+$ clones 1&3.

The SC101* contains a BglII site that the SC101 does not contain. This was used as a diagnostic marker for the correct origin on PCRs between primers 189 and 190. A gel showing this reaction for the consistent pMTS-*pR-tum*$^+$ and pMTS*-*pR-tum*$^+$ clones is shown in Figure 8.3. All consistent clones had the expected origins.

Sequencing was performed over the *pR-tum* section of all plasmids (between primers 329 and 457). All final plasmids also had the correct sequence over this region.



1–2. 3 µL digested origin PCRs of pMTS-*pR-tum*$^+$ C1&5 (expect 703 b.p.)
3–4. 3 µL digested ori PCRs of pMTS-*pR-tum*$^-$ C1&2 (expect 703 b.p.)
5. 500 ng 2-log ladder
6–9. 3 µL digested ori PCRs of pMTS*-*pR-tum*$^+$ C1,2,3&4 (expect 475 b.p. & 228 b.p.)

Figure 8.3: 2.0% agarose gel of diagnostic digests to check for the correct origin in the *pR-tum* plasmids.

8.3 Strains and DNA

8.3.1 Bacterial strains

Table 8.4: Bacterial strains used in this thesis.

Stock No.	Name	Genotype	Description
E2878	BW23473	Δ(*lacIZYA-argF*)U169 *rph-1* *rpoS396*(Am) *robA1 creC510* *hsdR514* Δ*endA9 recA1* *uidA*(Δ*Mlu*I)::*pir*(wt)	A *pir*$^+$ strain used for propagating and cloning plasmids with the *pir*-dependent R6Kγ origin. Obtained from B. Wanner [Haldimann and Wanner, 2001].
E2879	BW23474	Δ(*lacIZYA-argF*)U169 *rph-1* *rpoS396*(Am) *robA1 creC510* *hsdR514* Δ*endA9 recA1* *uidA*(Δ*Mlu*I)::*pir-116*	A *pir*$^+$ strain used for propagating and cloning plasmids with the *pir*-dependent R6Kγ origin. The *pir-116* mutation permits higher copy numbers. Obtained from B. Wanner [Haldimann and Wanner, 2001].
E4241	DH5α	*endA*1 *gln*V44 *thi*-1 *recA*1 *relA*1 *gyrA*96(Nal^R) *deoR* *nupG* Δ(*lacIZYA-argF*)U169 *hsd*R17($r_K^-m_K^+$), λ^-, F^-	General cloning strain (Bethesda Research Laboratories).
E4300	NK7049	Δ*lac*χ74 *galOP*308 *rpsL* *su*$^-$	A K-12 *E. coli* strain used as the host for all reporter assays. Obtained from R. Simons [Simons et al., 1987].

8.3.2 Primers

Primers were reconstituted to 1 µg/µL in Tris-EDTA buffer and stored at −20 °C.

Table 8.5: Primers used in the course of this thesis. Sequences are written 5′ to 3′. Capitalisation indicates the intended priming region. Shaded boxes indicate locations of restriction enzyme sites.

No.	Sequence	Description
164	1 cggaag cttaag TTTAAC TTTAAG 25 AAGGAG A	Primes the RBS in pET3a plasmids (HindIII).
315	1 GCTAGT TATTGC TCAGCG GTGG	Reverse primer over the T7 terminator. It is located downstream of genes in, for example, the pET3a plasmid.
466	1 GGCATC ACGGCA ATATAC	Used with primers 469, 467 & 468 to check for single integrants at the λ attB site.
467	1 ACTTAA CGGCTG ACATGG	Internal primer for the CRIM plasmids. Used to check for single integrants at various attB sites.
468	1 ACGAGT ATCGAG ATGGCA	Internal primer for the CRIM plasmids. Used to check for single integrants at various attB sites.
469	1 TCTGGT CTGGTA GCAATG	Used with primers 466, 467 & 468 to check for single integrants at the λ attB site.
585	1 GGAATC AATGCC TGAGTG	Used with primers 586, 467 & 468 to check for single integrants at the HK attB site.
586	1 GGCATC AACAGC ACATTC	Used with primers 585, 467 & 468 to check for single integrants at the HK attB site.
706	1 gggggc ggccgc CTATTT CTTCCA 25 GAATTG C	Primes upstream of the SC101 or SC101* origins (NotI site).
707	1 gggggc tagcGA GAATCC AAGCAC 25 TAGG	Primes downstream of the SC101 or SC101* origins (NheI site).
754	1 gctcag tcatat ggtgAG CAAAGG 25 AGAAGA ACTTTT CACTGG	Primes N-terminal of folding reporter GFP (NdeI site).
755	1 gctcac gtacgT TATTTG TAGAGC 25 TCATCC ATGCC	Primes C-terminal of folding reporter GFP (BsiWI site).
813	1 gagttg gatccG GTTAAT TAACGG 25 CACCAC C	Primes upstream of the *tonB* terminator in pIT3-CL-*lacZ*flip (BamHI site).
814	1 gaggta cacggt gtgcca gcttct 25 gtatgg AAACCT CGCGCC TTACC	Primes downstream of the *tonB* terminator in pIT3-CL-*lacZ*flip (XcmI site).

8.3.3 Plasmids

Table 8.6: Plasmids referred to and cloned in the course of this thesis.

Name	Origin	Resist.	Description
pET3a-*c*I	ColE1	Amp	Source of *c*I gene with enhanced RBS for pZC320-ER-*c*I. Obtained from K. Shearwin. Construction is specified in Shearwin and Egan [1996].
pET-TumHis$_6$	ColE1	Amp	Original source of *tum*4$^-$5$^-$*his* gene for all MFL constructs. Obtained from K. Shearwin. Construction is specified in Shearwin et al. [1998]
pIT3-CL-*lacZ*flip	R6Kγ	Chlor	Source of *tonB* terminator for pZC320-*tonBterm* and *lacZ*O2$^-$ for pZC320t-*lacZ*. Obtained from I. Dodd. The *tonB* terminator and *lacZ*O2$^-$ gene in this plasmid are the same as in the pIT3-HFCL plasmid described in Cui et al. [2013], except that the reporter fragment between the two SphI sites has been flipped.
pET28-*gfp*	ColE1	Kan	Source of folding reporter GFP, which includes both the cycle 3 and enhanced GFP mutations, but was designed as a fusion protein to report on protein folding [Waldo et al., 1999]. Obtained from G. Waldo.
pSC101	SC101	Chlor	A medium-copy plasmid (10–12 copies) [Lutz and Bujard, 1997] and source of SC101 for pMCS-*pR*-*tum*$^+$. Obtained from H. Bujard.
pSC101*	SC101*	Chlor	A low-copy plasmid (3–4 copies) [Lutz and Bujard, 1997] and source of SC101* for pMCS*-*pR*-*tum*$^+$. Obtained from H. Bujard.
pZE55-λ *c*I	ColE1	Tet	Source of Tetracyclin (Tet) resistance gene for pMTS-*pR*-*tum*$^+$ plasmid. Obtained from I. Dodd.
pUHA-1	p15a	Kan	Supplies LacI repressor. Obtained from H. Bujard.
pZC320	mini-F	Amp	A low-copy plasmid backbone derived from the F plasmid that maintains close to chromosomal copy-numbers [Shi and Biek, 1995].
pZC320-WR-*c*I	mini-F	Amp	Supplies CI from the P_{lac} promoter with a wild-type RBS. Obtained from I. Dodd. This plasmid is identical to pZC320-186*c*I as specified in Dodd and Egan [2002].

Continued on next page

Table 8.6: *Continued from previous page*

Name	Origin	Resist.	Description
pZC320-ER-*c*I	mini-F	Amp	Supplies CI from the P_{lac} promoter with an enhanced RBS from the pET3a plasmid. Created by PCR amplifying the pET3a RBS and *c*I gene from pET3a-*c*I using primers 164 (adds HindIII site) and 315, and cloning between the HindIII and BamHI sites in pZC320.
pZC320-*tonBterm*	mini-F	Amp	Precursor to pZC320t-*lacZ*. Created by PCR amplifying the *tonB* terminator from pIT3-CL-*lacZ*flip using primers 813 and 814 and cloning between the BamHI and XcmI sites of pZC320, replacing the *lacZ*α fragment.
pZC320t-*lacZ*	mini-F	Amp	Induction system reporter used to quantify the ultrasensitivity in relieving LacI repression of P_{lac}. Created by cloning the *lacZ*O2^{-} fragment between the XhoI and BamHI sites of pIT3-CL-*lacZ*flip into the same sites of the pZC320-*tonBterm* plasmid.
pINT-ts	oriR101	Amp	CRIM helper plasmid for integration at the λ attB site [Haldimann and Wanner, 2001]. Obtained from B. Wanner. Strains with this plasmid must be grown at 30°C due to the temperature-sensitive origin.
pAH69	oriR101	Amp	CRIM helper plasmid for integration at the HK attB site [Haldimann and Wanner, 2001]. Obtained from B. Wanner. Strains with this plasmid must be grown at 30°C due to the temperature-sensitive origin.
placatt1-Δ*lacY-lacZ*	R6Kγ	Spec	β-gal reporter chassis that can be integrated at the λ attB site using the CRIM system. Obtained from I. Dodd. Details of its genetic features are given in Section 3.1.1 and Figure 3.2(a).
pIT-SL-*pR-lacZ*	R6Kγ	Spec	Contains the short *pR*–*pL* fragment of phage 186 reading from *pR* into *lacZ*. Used to generate *pR-lacZ* reporter strains by integrating into the λ attB site. Derived from placatt1-Δ*lacY-lacZ*. Obtained from M. Pocock. Details of its genetic features are given in Section 3.1.1 and Figure 3.2(a).

Continued on next page

Table 8.6: *Continued from previous page*

Name	Origin	Resist.	Description
pIT-CH-*pR-tum*$^+$	R6Kγ	Chlor	Contains the short *pR–pL* fragment of phage 186 reading from *pR* into *tum*4$^-$5$^-$*his*. Used for cloning pIT-CH-*pR-gfp*. Obtained from M. Pocock, who derived pIT-CH-*pR-tum*$^+$ from placatt1-Δ*lacY-lacZ* by first replacing *lacZ* with *tum*4$^-$5$^-$*his* from pET-TumHis$_6$, then replacing the Spec resistance gene with a Chlor resistance gene, and finally replacing λ attP with the HK attP from the pAH144 CRIM plasmid.
pIT-CH-*pR-gfp*	R6Kγ	Chlor	Contains the *pR* promoter reading into *gfp*. Used to generate *pR-gfp* reporter strains by integrating into the HK attB site. Created from pIT-CH-*pR-tum*$^+$ by amplifying the folding reporter *gfp* gene from pET28-*gfp* using primers 754 and 755, and cloning between the NdeI and BsiWI sites of pIT-CH-*pR-tum*$^+$ to replace the *tum*4$^-$5$^-$*his* gene. Primer 754 adds an initiation codon and a valine codon to the fusion protein *gfp* template.
pMCS-*pR-tum*$^+$	SC101	Chlor	A medium-copy plasmid with the *pR* promoter reading into *tum*4$^-$5$^-$*his*. Precursor to pMTS-*pR-tum*$^+$. Obtained from M. Pocock, who derived pMCS-*pR-tum*$^+$ from pIT-CH-*pR-tum*$^+$ by cloning the fragment between the NotI and NheI sites of pSC101 into pIT-CH-*pR-gfp*, replacing the R6Kγ origin.
pMCS*-*pR-tum*$^+$	SC101*	Chlor	A low-copy plasmid with the *pR* promoter reading into *tum*4$^-$5$^-$*his*. Precursor to pMTS*-*pR-tum*$^+$. Obtained from M. Pocock, who derived pMCS*-*pR-tum*$^+$ from pIT-CH-*pR-tum*$^+$ by cloning the fragment between the NotI and NheI sites of pSC101* into pIT-CH-*pR-gfp*, replacing the R6Kγ origin.
pMTS-*pR-tum*$^+$	SC101	Tet	The medium-copy MFL plasmid with *pR* reading into *tum*4$^-$5$^-$*his*. Created by three fragment ligation of SpeI/AatII digests of the pMCS-*pR-tum*$^+$ plasmid and the pZE55-λ *c*I plasmid. Gel extraction of the required fragments was used to maximise success. Additional details for preparing this clone are given in Section 8.2.9.

Continued on next page

Table 8.6: *Continued from previous page*

Name	Origin	Resist.	Description
pMTS*-*pR-tum*$^+$	SC101*	Tet	The low-copy MFL plasmid with *pR* reading into *tum*4$^-$5$^-$*his*. Created by three fragment ligation of SpeI/AatII digests of the pMCS*-*pR-tum*$^+$ plasmid and the pZE55-λ *c*I plasmid. Gel extraction of the required fragments was used to maximise success. Additional details for preparing this clone are given in Section 8.2.9.
pMTS-*pR-tum*$^-$	SC101	Tet	The medium-copy *tum*$^-$ MFL control plasmid. Created by digesting pMTS-*pR-tum*$^+$ with XhoI and SalI, which have compatible ends, and religating a dilute solution of the digest to favour intramolecular ligation.
pMTS*-*pR-tum*$^-$	SC101*	Tet	The low-copy *tum*$^-$ MFL control plasmid. Created by digesting pMTS*-*pR-tum*$^+$ with XhoI and SalI, which have compatible ends, and religating a dilute solution of the digest to favour intramolecular ligation.
pIT-SL	R6Kγ	Spec	Empty integration chassis. Created by digesting placatt1-Δ*lacY-lacZ* with XhoI and SalI, which have compatible ends, and religating a dilute solution of the digest to favour intramolecular ligation.

8.4 Assays

8.4.1 Preparation of cell extracts

For analysis of cell protein content by Western blot, cell extracts first need to be prepared. The base protocol starts from 2 mL O/N cultures of the required strain, which are subcultured 1/200 into 4 mL of pre-warmed LB and grown to log phase with shaking at 37°C. During growth to log phase, the OD_{600} of the cultures was monitored by taking 100 µL samples into the wells of a 96-well plate, determining optical densities using the plate reader as previously described (Section 8.2.1). Upon reaching OD_{600} 0.6, cultures were placed on ice for 30 minutes. After resuspension, a final reading of OD_{600} was made and used to normalise aliquots of the cultures into 1.5 mL screw-capped microcentrifuge tubes to obtain the equivalent of 900 µL of culture with OD_{600} 0.6. These samples were pelleted (10 minutes, 13000 rpm, 4°C, Eppendorf Microcentrifuge 5418R) and the supernatant carefully removed using a drawn-out Pasteur pipette connected to a vacuum source. After re-pelleting (1 minute, 13000 rpm, 4°C) and removing remaining supernatant, pellets were resuspended in 40 µL of Lysis Solution (200 ng/mL lysozyme and 250 U/mL benzonase

made up fresh in B-PER Reagent and kept on ice). Lysis proceeded on ice for 30 minutes, after which 40 µL of 2× Extract Loading Buffer (2× NuPAGE LDS sample buffer (Invitrogen) and 2× NuPAGE sample reducing agent (Invitrogen) made up fresh in H_2O) were added. Finally, samples were heated at 70°C for 10 minutes to denature the protein and pulse spun to recover condensation. These samples were stored at −20°C until required.

The base protocol was used to prepare 1× extracts for quantitation of steady-state protein levels. Variations on the base protocol were used to prepare extracts for serial dilutions, or quantitation of protein degradation or protein solubility. For quantitation of steady-state protein levels, extracts of cells not containing the target protein were used for serial dilutions of the samples and the protein standard. This was done to ensure that any background bands in the Western blots would be consistent between samples. These extracts were prepared in large scale.

Large scale protocol. This is essentially a scaled version of the base protocol. Log phase cultures were instead 50 mL. Final aliquots were normalised to obtain the equivalent of 9 mL of culture with OD_{600} 0.6. Cells were pelleted in 10 mL yellow-capped tubes using an Eppendorf Centrifuge 5810R (15 minutes, 4000 rpm, 4°C in the first spin; 5 minutes in the second). Cells were resuspended in 400 µL of 2× Lysis Solution in the yellow-capped tubes, and were only aliquotted into 1.5 mL screw-capped tubes after mixing with 400 µL of 2× Extract Loading Buffer. Protein denaturation and storage were as for the base protocol.

Degradation extracts. The preparation of extracts to measure protein degradation proceeded similarly to the large scale protocol. Log phase cultures were again 50 mL, however, about 10 minutes before reaching OD_{600} 0.6, $MgSO_4$ was added to 50 mM from a stock of 1 M to reduce loss of macromolecules and cell lysis when inhibting translation [Gupta, 1975]. After 10 minutes, protein production was stopped by adding chloramphenicol to 200 µg/mL (from a 100 mg/mL stock made up fresh in ethanol); cultures continued to be maintained at 37°C with shaking. At regular intervals, 1 mL samples of the culture were taken into 5 mL of ice-cold 1× PBS and left on ice for 30 minutes. The chilled samples were pelleted as for the large scale protocol, but after removal of all supernatant, a freeze/thaw step (10 minutes at −80°C) was applied to enhance lysis. Resuspension in Lysis Solution and mixing of Extract Loading Buffer were again performed in the yellow-capped tubes, though for volumes of 40 µL each.

Soluble and insoluble fraction extracts. The preparation of soluble and insoluble fraction extracts proceeded similarly to the base protocol. After pelleting of cells and removal of all supernatant, a freeze/thaw step (10 minutes at −80°C) was applied to enhance lysis. Pellets were resuspended in 100 µL of ice-cold Lysis Solution and left on ice for 30 minutes. 40 µL and 50 µL aliquots were taken into 1.5 mL screw-capped microcentrifuge tubes, the

former being kept aside as the 'total protein' sample. The latter was spun for 10 minutes at 13000 rpm and 4°C in an Eppendorf Microcentrifuge 5418R, and 40 μL of the supernatant aliquotted into another tube (the 'soluble fraction'). The remaining supernatant (S/N) was carefully removed using a drawn-out Pasteur pipette connected to a vacuum source, and the pellet resuspended in another 50 μL of Lysis Solution (the 'insoluble fraction'). Addition of Extract Loading Buffer, protein denaturation and storage of all samples (total protein and soluble and insoluble fractions) were as for the base protocol.

Pre-adsorption extracts. Pre-adsorption extracts were used to quench any non-specific antibodies present in the primary antibody preparations. These were prepared from 500 mL log phase cultures (OD_{600} 0.6) of cells not containing the target proteins of interest. After pelleting by centrifugation, the culture was resuspended in 5 mL of 10 mM Tris-HCL pH 8.5 and 1 mM $MgCl_2$. The resuspension was sonicated with 2–4 10 second pulses, treated with 5 μL of benzonase (25 U/μL) and left on ice for 30 minutes, then dispensed into 1.5 mL screw-capped tubes for storage at −20°C.

8.4.2 Polyacrylamide gel electrophoresis of proteins and Western blotting

The protein content of cellular extracts was quantified by first separating on the basis of molecular weight by SDS-PAGE, then transferring by Western blot and probing for the protein of interest by antibody staining.

SDS-PAGE was performed using the Xcell Surelock Mini-Cell system with precast NuPAGE Novex 12% Bis-Tris 12-well minigels (Invitrogen). Gels were run according to the manufacturer's specifications at 200 V with the NuPAGE MOPS SDS Running Buffer, loading 12 μL of sample cell extract per well. When needed, ECL Plex Fluorescent Rainbow Markers (Amersham) were added in one or more of the wells to act as a molecular weight standard.

Proteins were transferred from completed gels to Amersham Hybond-LFP membranes (low fluorescence PVDF membranes with a 0.2 μm pore size) using the XCell II Blot Module for the Surelock Mini-Cell system. Transfers were run according to the manufacturers instructions using NuPAGE Transfer Buffer and constant 30 V for 1 hour.

Membranes were blocked overnight in 5% BSA in PBS with shaking at 4°C. Membranes were then rinsed twice in PBS-T (1× PBS with 0.1% tween), and washed twice in PBS-T for 5 minutes with shaking at room temperature. The membrane is then incubated with 20 mL of pre-adsorbed primary antibody (a 1/500 or 1/1000 dilution of primary antibody in PBS-T incubated with 300 μL of pre-adsorption extract for 30 minutes) for 1.5 hours at room temperature. It is again washed several times with PBS-T before incubating with secondary antibody (20 mL of a 1/5000 dilution of ECL Plex Goat-α-Rabbit IgG-Cy5 (Amersham) in PBS-T) for 1 hour. Finally, membranes were washed several times in PBS-T, then several times in PBS, before being dried between blotting paper for 1 hour at 37°C.

Membranes were scanned on a Typhoon Imager (GE Healthcare) using the 633 nm laser and 670/30 bandpass filter using a low to moderate Photomultiplier Tube (PMT) voltage (∼500 V).

Quantitation of band intensity from the scanned images was performed using the `ImageJ` application [Rasband, 2011]. After 'rolling ball' background subtraction (a method that accounts for local variations in background intensity) the intensity of bands was measured as the total intensity within a rectangular selection encompassing each band.

8.4.3 Quantitating concentrations of cells in culture

To normalise the quantitated levels of protein in cell extracts to a value per cell, the concentration of cells in cultures of OD_{600} 0.6 were measured. To match the cell extract protocol as closely as possible, log phase cultures were grown to around OD_{600} 0.6 and the OD_{600} of 100 µL samples then more precisely measured using the plate reader as previously described (Section 8.2.1). Using these measured OD_{600}s, a normalising culture volume was derived to obtain the equivalent of 10 µL of culture with OD_{600} 0.6. This calibrated volume of culture was added to 990 µL of LB and then further serially diluted (by steps no greater than a 1/100 dilution) to obtain countable numbers of cells (final dilution factors were in the range 1×10^6 to 1×10^7). 100 µL of this final dilution was very gently spread onto L-plates with appropriate antibiotics until most of the liquid had been absorbed; the remaining liquid was allowed to absorb without spreading to avoid killing any cells. After growth overnight at 37°C, colonies were manually counted on each plate.

8.4.4 Growth of bacteria for 96-well plate assays

Bacteria grown to log phase in 96-well plates were used for LacZ assays and flow cytometry assays. The methods detailed here are based on ones described in Dodd et al. [2001] and Palmer et al. [2009] for the growth of bacteria for LacZ assays.

Basic assay. Growth was started from colonies on plates freshly streaked from glycerol stocks. Yellow pipette tips were used to pick colonies and suspend them in 98 µL of growth medium in the wells of a 96-well plate. 2 µL of this suspension was then pipetted into new wells containing 98 µL of growth medium for growth overnight at 37°C. To minimise evaporation of the growth medium, ethanol-swabbed rubber-padded lids were used and plates were also sealed with sticky tape.

The following day, log phase cultures were established from the overnight cultures. All dilution steps were performed at 37°C. First, the overnight cultures were diluted 1/5 into 100 µL of growth medium in new wells and the A_{620}s measured using a Multiskan Ascent Plate Reader (Labsystems). The A_{620}s were converted to OD_{600}s using an empirically determined relationship [Dodd et al., 2001]. These values were used to calculate normalising volumes of growth media for each culture, to which 50 µL of the diluted overnight culture

could be added to reach an effective final OD_{600} of 0.15 (typically giving an overall 1/10 dilution relative to the original overnight culture). These normalised cultures were then diluted 1/50 into 100 μL of growth medium in the wells of a new 96-well plate. Typically at this point, each normalised culture was used to innoculate multiple wells that differed only by the concentration of IPTG in the growth medium. These cultures were grown to late log phase (A_{620} of 0.13–0.17) in a custom-made rotating drum for 96-well plates. Growth was followed using the plate reader, and for growth assays, these measurements were made at regular intervals. Upon reaching late log phase, cells were ready for assay.

For a given type of assay, stocks of growth media with appropriate antibiotics and a range of IPTG concentrations were prepared in advance and stored in aliquots at $-20°C$. These were thawed and pre-warmed to 37°C when needed. After thawing, any unused media was disposed of. Unless otherwise specified, the overnight cultures and dilutions thereof were performed in growth media without IPTG. Growth media with a range of IPTG concentrations was typically used only in the final log phase plate.

Hysteresis assays. The short equilibration time hysteresis assay proceeded similarly to the basic assay, however, each colony suspension was used to innoculate two alternative overnight cultures: one with 0 μM IPTG, the 'low O/N', and one with 300 μM IPTG, the 'high O/N'. These were diluted and normalised the following day using media containing consistent IPTG concentrations. After normalisation, 2 μL aliquots of each culture were added to wells of the final log phase plate containing 98 μL of growth medium with a range of different IPTG concentrations.

The normal equilibration time hysteresis assay introduced an additional 1/10 dilution of the overnight cultures (both low and high) in 100 μL of growth media with consistent IPTG concentrations. This step was performed directly after the normalisation step, so the further-diluted cultures were the source of aliquots for the log phase plate.

The long equilibration time hysteresis assay proceeded as for the short equilibration time hysteresis assay, except that once the cultures had reached an OD_{600} 0.3 during growth to log phase, the entire plate was subcultured 1/50 into a new log phase plate with an identical arrangement of growth media.

The very long equilibration time hysteresis assay combined both normal and long equilibration time variations on the short equilibration time assay.

8.4.5 LacZ assay

The LacZ assay protocol was based on that described in Dodd et al. [2001] and Palmer et al. [2009]. Cells were grown to late log phase in 96-well plates as described in Section 8.4.4. Growth was quickly stopped by setting the plate on a pre-chilled metal block in ice for 15 minutes. 20 μL of each culture was then transferred to another 96-well plate, pre-heated to 30°C, and containing 150 μL of TZ8+ buffer and 30 μL of LB per well. After

a 15 minute incubation at 30°C to permit lysis, 40 µL of ONPG solution (4 mg/mL ONPG in TZ8 buffer) was added to each well to start the assay. The rate of cleavage of ONPG by β-gal was followed by measuring the increase in absorption of each well at 414 nm in a Multiskan Ascent plate reader (Labsystems) set to hold the temperature at 28°C. Measurements were made every 2 minutes for 1 hour. LacZ units are calculated from the slope (M, per minute) of this line and the final OD_{600} (O) of the log phase culture using the formula:

$$U = 1000 \times \frac{M \times V_r}{O \times V_c} \tag{8.1}$$

where V_r and V_c are respectively the reaction volume before addition of ONPG (200 µL) and the volume of culture used (20 µL). Due to a short lag time at the beginning of the assay and saturation in absorption for wells with very high activity, some points are systematically omitted before calculating the slope to capture only the window where the increase in absorption is linear.

Calculation of LacZ units and statistical analysis of the data was performed using R [R Development Core Team, 2012].

8.4.6 Flow cytometry

For assay by flow cytometry, cells grown to late log phase as described in Section 8.4.4 were first stopped by setting the plate on a pre-chilled metal block in ice for 15 minutes. 40 µL of each culture was then transferred into the empty wells of a new 96-well plate and the cells pelleted in that plate by centrifugation (1300 × g for 5 minutes at 4°C in an Eppendorf Centrifuge 5810R). The supernatant was removed from each well using a drawn-out Pasteur pipette connected to a vacuum source, and the cells then resuspended in 80 µL of ice-cold 1× PBS using a 30 second shake protocol in the Multiskan Ascent plate reader to facilitate resuspension. The cells were again pelleted (1300 × g for 5 minutes at 4°C), the supernatant was removed and the cells resuspended in 40 µL of ice-cold 1× PBS with shaking. 1/20 dilutions of this cell suspension were made to 300 µL in ice-cold 1× PBS in round-bottom 12 × 75 mm tubes.

These samples were run through a FACSCanto Flow Cytometer (Becton Dickson) according to the manufacturer's specifications. The cytometer was configured with a Coherent® Sapphire™ Solid State laser for excitation at 488 nm and a 530/30 bandpass filter to record fluorescence emission due to GFP.

For the low cell count assays, around 1×10^4 to 3×10^4 cells were measured at a medium flow rate, giving 500–3000 events per second. PMT voltages were set at 600 V, 600 V and 800 V for FSC, SSC and fluorescence channels respectively. The threshold for recorded events was set so that both FSC and SSC needed to be larger than 2×10^4, which ensured exclusion of debris whilst retaining events from the main bacterial population. Peak area, peak height and peak width were recorded for each channel.

For the high cell count assays, around 8×10^4 to 1×10^5 cells were measured at a medium flow rate, giving 1000–6000 events per second in this case. PMT voltages were set at 350 V, 450 V and 750 V for FSC, SSC and fluorescence channels respectively. The threshold for recorded events was set so that both FSC and SSC needed to be larger than 300. Peak area, peak height and peak width were recorded for each channel.

Basic filtering and analysis was done in R using the `flowCore` [Ellis et al., 2009a] and `flowClust` [Lo et al., 2009] packages from BioConductor [Gentleman et al., 2004]. Custom scripts were used for final analysis as described in Appendix B.

9

Structure-function studies for Tum

With the aim of increasing the utility of the Tum protein as a new module for synthetic biology, a draft manuscript was prepared that presents new work on characterising Tum through a collection of Tum mutants. It was written to be submitted into the journal PLoS Genetics.

I cloned some of the truncation mutants and performed LacZ assays for all of the truncation mutants. I performed the data analysis (including the Hill curve fits of the data, the alignment of Tum homologs, and the structure prediction efforts) and wrote the manuscript.

This draft manuscript has not yet been submitted to any journal.

Statement of Authorship

Title of Paper	Mapping function to structure in the Tum antirepressor
Publication Status	○ Published, ○ Accepted for Publication, ○ Submitted for Publication, ◉ Publication style
Publication Details	This draft manuscript reports on a number of mutants obtained for the Tum antirepressor of phage 186 and makes use of this data and an increasing wealth of bioinformatic data to make functional and structural predictions regarding this protein.

Author Contributions

By signing the Statement of Authorship, each author certifies that their stated contribution to the publication is accurate and that permission is granted for the publication to be included in the candidate's thesis.

Name of Principal Author (Candidate)	Julian Pietsch		
Contribution to the Paper	Cloned some of the truncation mutants and performed LacZ assays for all of the truncation mutants; performed data analysis, including the Hill curve fits, discovery and alignment of the Tum homologs, and structure prediction; wrote the manuscript.		
Signature		Date	December 17 2014

Name of Co-Author	Ian Dodd		
Contribution to the Paper	Designed research; performed research; contributed new reagents/analytic tools.		
Signature		Date	December 17 2014

Name of Co-Author	Keith Shearwin		
Contribution to the Paper	Designed research; performed research; contributed new reagents/analytic tools.		
Signature		Date	December 17 2014

Name of Co-Author			
Contribution to the Paper			
Signature		Date	

Mapping function to structure in the Tum antirepressor

Julian M. J. Pietsch[1], Ian Dodd[1], Keith E. Shearwin[1,*]
1 Department of Biochemistry, University of Adelaide, Adelaide, Australia
∗ E-mail: keith.shearwin@adelaide.edu.au

Abstract

In an apparent effort to maximise host success and fast colonisation, temperate bacteriophages choose between two modes of development: lytic growth where the host is coerced into mass production of infectious phage particles, or lysogenic growth where the phage chromosome is integrated within its host's. During lysogeny, a repressor protein keeps lytic genes silent and the transition from lysogeny to lytic development requires relief of this repression. In many cases, this occurs in response to stress signals such as the host SOS pathway, invoked under conditions of DNA damage. An emerging mechanism of phage sensitivity to this pathway is the upregulation of a phage-borne antirepressor protein that directly inactivates the lytic repressor. Here we develop assays to quantitatively characterise the *in vivo* antirepressor function of the Tum protein from coliphage 186. We generate libraries of Tum mutants using complementary approaches and screens designed to favour functionally relevant mutations. By combining a detailed analysis of these mutants with sequence features identified for Tum and its homologs, we show that the Tum protein consists of two domains: an amino-terminal domain (NTD) that is responsible for antirepressor activity and is representative of a broad family of little-studied small phage proteins, and a carboxy-terminal domain showing strong homology with DinI-like genes seen in many other phage. The highly ultrasensitive antirepressor activity can be further mapped to two extended α-helices in the NTD. Efforts at experimental structure determination have been unsuccessful so we instead used our mutational data set to assist the *ab initio* prediction of valuable structural models. These models in turn complement the analysis of the mutation data and provide functional insights into the activity of the antirepressor domain.

Author Summary

As highly mobile vectors for DNA transport, bacteriophage play a significant part in the fast evolution and adaptation of bacterial populations to new environments. In temperate bacteriophage, this spread of genetic information is modulated by the well-studied lysis-lysogeny decision, where the phage choose between two growth modes by a complex integration of host and environmental signals. In lysogeny, the phage genome is integrated within the host's and a repressor protein typically plays an essential role in keeping the lytic genes silent. Relief of this repression results in lytic development, and numerous pathways exist to effect such development. Small antirepressor proteins are emerging as an widely-used mechanism for such phage induction, often also implicated in trans-phage activation. Here we report on a collection of mutations obtained for the Tum antirepressor of phage 186, and make use of this data and the increasing wealth of bioinformatic data to dissect the function of these powerful but small proteins. By combining the different data sets we have been able to build predictive structural models.

Introduction

Microbial ecosystems thrive on sharing genetic information. Driven by a need for fast evolution and adaptation, numerous mechanisms are employed to modulate and enhance this process [1]. Within this dynamic world, viruses and bacteriophage play a major role as highlighted by the abundant occurrence of genes in the viral metagenome that enhance host fitness [2]. Temperate bacteriophage, in particular, are well suited to mediating genetic transactions given their two distinct life cycles: during lysogeny the

phage genome is integrated as part of the host chromosome, silently replicating with the host, whilst in lytic growth, the phage turns its host towards mass production of phage, inducing cell lysis and the consequent release and spread of more infectious particles. In this part-symbiotic role, temperate phage are often observed to harbour 'morons' — genes that have no function in phage development, but are co-transported and often confer competitive advantages to the host [3]. An especially consequential subset are morons that increase host pathogenicity, well-known examples including the phage-derived genes for the cholera [4] and Shiga [5] toxins.

The balance between lysis and lysogeny bears on the proliferation potential of temperate phage, and in consequence the decision between the two growth modes is highly regulated. A classic example is induction of the λ coliphage as a function of the host SOS response, a well-conserved DNA repair pathway. Genes involved in the SOS response are normally held silent by the host LexA repressor, but are derepressed when autocatalytic cleavage of LexA is stimulated by the formation of RecA protein filaments along single-stranded DNA [6]. By utilising a similar RecA-dependent self-cleavage mechanism, the CI repressor that maintains lysogeny in phage λ is also inactivated upon SOS induction [7]. In contrast, a number of other phage respond to this same pathway by harbouring LexA-repressible promoters in their genomes. These promoters may directly control lytic transcripts, as in the phage responsible for cholera [4], or promote lytic development via the induction of intermediates known as 'antirepressors': phage proteins which can inactivate the repressors responsible for maintaining lysogeny. This latter mechanism also provides the capacity for target promiscuity, enabling the coordinated induction of disparate prophage by the evolution of repressors that respond to the same antirepressor [8]. Ironically though, this mechanism has also seen the phage themselves exploited: Staphyloccocal Pathology Islands (SaPIs) respond to phage induction by employing repressors that respond to antirepressors of helper phage. The SaPIs then hijack phage capsid production for the transport of their own large islands of pathogenic genes [9]. These findings have revealed an important new layer in phage interaction and cooperation that is mediated by these antirepressor proteins.

The P2-related bacteriophage 186 stands as one of the first studied examples of phage induction via an antirepressor [10]. In this phage, the CI repressor maintains lysogeny by holding the lytic promoter p_{R} off (see Figure 1), but unlike λ CI, the 186 repressor is not sensitive to RecA-facilitated cleavage [11]. Instead, the phage harbours an 'SOS operon' that is transcribed from the LexA-repressible promoter p_{95} and consists of two genes: the antirepressor *tum*, which is responsible for derepression of the lytic p_{R} promoter and induction of the phage, and *orf97*, which is less well-characterised but whose overexpression grants immunity to further infection by 186 virions [12]. The Tum antirepressor acts reversibly against the CI repressor by preventing it from binding to DNA, thereby promoting lytic development [13]. The carboxy-terminal (C-terminal) region of Tum shows homology with *dinI*-like genes found in many other prophage [14], but in these cases, it is an independently translated gene product. Homologs of the amino-terminal (N-terminal) region are similarly found as independently translated genes [15].

Here we define and characterise the region of Tum that is responsible for its antirepressor activity. A bioinfomatic assessment of Tum and its homologs strongly suggests a two-domain protein and reveals a family of little-studied homologs with the putative N-terminal domain. Using a quantitative *in vivo* assay for antirepressor activity to characterise a broad assortment of Tum mutants, we confirm that the Tum protein has two domains with the N-terminal domain being responsible for its antirepressor activity. Truncation mutations reveal that the first 44 amino acids are sufficient for activity and this small modelling region forms the basis for building structural models of Tum that refine our understanding of this class of antirepressor.

Results

Sequence analysis

Bacteriophage 186 is classed as a member of the P2 family, and is most closely related to the PSP3, Fels-2 and SopEΦ phages of *Salmonella* [16]. Each of these phages have 186 *tum* homologs [15–17], but strikingly, the region of *tum* homology in each of the homologs is split into two separately translated genes (see Figure 2). A search of the NCBI's non-redundant protein sequence database ('`nr`') for homologs spanning the full length of Tum resulted in only 10 hits using the PSI-BLAST program [18], all showing high sequence similarity with Tum (ranging from 60%–97% identity) and found variously in prophages occupying *Salmonella*, *Klebsiella*, or *Enterobacter* strains. More than 200 other hits were identified that were homologous to subregions of Tum, and these were clearly split into two groups, one aligning within the first 70 amino acids, and the other aligning with amino acids 71–146. The latter formed the vast majority and were ascribed to the DinI superfamily.

Independently translated DinI-like genes are frequently found within the SOS operons of many phages [14], including the downstream Tum-like genes in the PSP3, Fels-2 and SopEΦ phages. Intriguingly, in addition to its full length product, the *tum* gene also produces three alternative translation products from internal translation start sites [12], and the shortest of these, open reading frame 5 (ORF5), exactly matches the C-terminal homology region (see Figure 2). However, none but the full-length ORF are capable of phage induction, suggesting that the antirepressor activity of Tum is a function of the N-terminal region. This is supported by preliminary evidence suggesting that the antirepressor activity is associated with the upstream and not the downstream gene in the PSP3 phage [15].

To better characterise the smaller family of N-terminal sequence homologs, a seed alignment of residues 1–70 of Tum with its respective PSP3, Fels-2 and SopEΦ homologs was submitted to the HHsenser server [19], searching again over the '`nr`' database. HHsenser is designed to find diverse homologs by performing a recursive PSI-BLAST search from 'seed' sequences outside the standard PSI-BLAST threshold, so that each seed represents a homology group. 56 seed sequences were identified apart from the query sequences, this set of sequences standing representative for a group of about 240 N-terminal Tum homologs in the '`nr`' database. The resulting alignment of those sequences is depicted in Figure 2 for the query sequences only, with the degree of conservation across the entire set summarised by alignment quality.

The regions of high conservation are limited to the two extended N-terminal α-helices as predicted from the HHsenser search. Most of the well-conserved residues are hydrophobic and likely to form the stable hydrophobic core for a putative Tum amino-terminal domain (NTD). Of particular note, the negatively charged residues at positions 30 and 40 are highly conserved throughout the family and are indicative of the overall negative charge of this region. Curiously, a cysteine in a putative loop between the two helices was also present in 52 of the 58 seed sequences, an especially high identity compared with the neighbouring residues. For comparison and reference, a search for homologs to the C-terminal region of Tum was similarly performed and the resulting alignment with *Escherichia coli* (*E. coli*) DinI and the phage homologs is shown in Figure 2.

Assaying Tum activity *in vivo*

The antirepressor activity of Tum was previously characterised *in vitro* in a gel shift assay that measured the reduction in CI–DNA binding with increasing Tum concentration [13]. We designed an analagous *in vivo* antirepressor activity assay that reports on relief of CI repression at the p_R promoter by an inducible level of Tum (see Figure 3A). Thus antirepressor activity is indirectly measured as the fraction of CI that remains free to repress p_R. UV-induction of the prophage normally requires Tum to operate against a lysogenic level of CI, and this guided our design. A single-copy P_{bla} promoter was used to provide lysogen-like levels of CI, giving 75-fold repression of p_R compared with 400-fold repression seen with a 186 prophage [20]. *In trans* constitutive production of CI avoids the natural feedback on CI expression

present in the phage [21]. To measure the fraction of free CI, a relatively large fragment of the phage 186 p_R–p_L switch region that includes the flanking F_L and F_R CI binding sites was cloned upstream of a *lacZ* reporter gene. This was chosen to better reproduce the extent of repression found in the prophage, since these flanking sites enhance CI repression of p_R [20].

The response of this assay strain to increasing levels of wild-type Tum is shown in Figure 3B. The Tum$^-$ control (a parental induction plasmid) shows the repression by production of CI from P_{bla}. The CI$^-$ control shows the maximum rate of production possible from unrepressed p_R, and also confirms that Tum by itself is unable to modulate p_R activity [13]. With sufficient Tum antirepressor, close to complete relief of p_R repression can be achieved. A Hill fit of the wild-type response reveals that it is highly ultrasensitive with an apparent Hill coefficient of 8 ± 2 — much higher than that observed *in vitro* [13].

The *in vivo* assay system also enabled direct comparison of the antirepressor activities of some *tum* variants. Silent mutations to the *tum* gene can be made to inactivate both the ORF4 and ORF5 internal ribosome binding sites [13]. This *tum*4$^-$5$^-$ gene produces overlapping antirepressor activity with wild-type *tum*, showing that products from these alternative ORFs do not interfere with antirepressor activity. Addition of a C-terminal His_6 tag makes no discernible difference to activity either, validating the use of *tum*4$^-$5$^-$*his* in the *in vitro* gel shift assay system [13].

Point mutants are found exclusively in the NTD

Many point mutations that result in complete loss of function are diagnostic for functional regions and critical active site residues of a protein. These are typically hidden amongst a milieu of less informative mutations that effect gross changes to protein stability or translation (for example, mutations to premature stop codons). Here we used a screening strain to select for point mutants of *tum* that are specifically unable to relieve CI repression, but which are not misfolded, frameshifted or truncated.

We selected for antirepression-deficient Tum mutants by screening a point mutant library (generated by error-prone PCR over the entire *tum* gene) *in trans* using a Prophage of Death (POD) strain that contains a 186 lysogen with a non-functional integrase (see Figure 4A). Without an integrase, the prophage cannot excise or produce functional progeny, so induction of the lytic pathway results in cell death [22]. Cells expressing Tum mutants will only survive if they have lost the ability to induce the phage. To exclude mutants that are inactive due to poor folding, frameshifts or truncations, we used a *tum*4$^-$5$^-$ template with a C-terminal fusion to the α-fragment of the *lacZ* gene (LacZα). In strains containing the complementary LacZω, only those mutants that successfully translate the full length of *tum*4$^-$5$^-$α and are not grossly misfolding [23] will produce a functional LacZ enzyme and hence the characteristic dark blue on plates containing X-gal.

A total of 50 blue colonies were selected for sequencing over the *tum*4$^-$5$^-$α gene. All contained point mutations at one of 18 distinct positions in the N-terminal region of Tum between residues 12–22 and residues 30–41, fitting within the two predicted N-terminal α-helices (see Figure 4B).

Four of the point mutants resulted in early stop codons, which were all located upstream of the ORF4 start codon. These truncation mutants will be presented later.

Quantitation of the mutants was done as for the wild-type *tum* constructs. To simplify comparison, we report antirepressor activity for the mutants as the summed area under each assay curve, shown normalised to the Tumα control in Figure 4B.

All of the point mutants obtained in the screen showed less than 25% the activity of the Tumα control, with most resulting in activities indistinguishable from the Tum$^-$ control (note that the two mutations, D2N and T21S, showing full activity were recreated from double mutants with the R19H and D30G mutations respectively). The majority of the mutations occurred at hydrophobic residues with three charge reversal mutations among those at charged and polar residues. The high density of strongly inactivating mutations in such a small region of the protein is good evidence for a structurally-isolated antirepressor function.

The collection of point mutants obtained in this study is complemented by the E12K, A34V and E40K mutations obtained in an earlier study using phage-based screens for loss of UV-inducibility [12]. Point mutations at these residues were also represented in the present screen, though the specific A34V and E40K mutations did not arise.

Mapping structured regions by insertion mutagenesis

Insertions of short stretches of amino acids throughout a protein can be used to map the tolerance of each region to structural disruption. Insertions in highly-structured regions, or buried inside a fold are more likely to disrupt protein function, whilst insertions in linker regions and surface-exposed loops are less likely to disrupt function. We prepared a library of insertions of 15 base pairs throughout *tum* using an *in vitro* transposase reaction on PCR-amplified *tum*4^-5^-*his* DNA. With an aim to obtaining a collection of insertions in both structured and unstructured regions of the protein we screened this library for mutants with a broad range of activities using an antirepressor activity reporter strain plated on X-gal. Clones within the library were not guaranteed independent insertion events, since an intermediate step involved pooling colonies to remove the antibiotic resistance cassette used for selecting the insertions. As a result, many of the clones were found multiple times.

The insertion event can occur in 6 different frames of the protein (counting forward and reverse orientations of the transposon), and results in insertion of 5 codons, with two of the six frames giving rise to stop codons. This was reflected in the library, where roughly one third of the *tum*4^-5^-*his* insertion mutants were found with introduced stop codons. The results of these truncation mutants will be presented in the following section.

As done for the point mutants, antirepressor activity assays were performed for each of the insertion mutants and the normalised areas calculated for these curves are shown in Figure 5 along with the insertion location and residues inserted. Consistent with the point mutations, insertions in either of the first two α-helices caused complete loss of activity. However, all other insertions retained at least 40% of wild-type activity, including three located between the first two α-helices, suggesting a flexible loop between these helices.

Truncation mutations confine the region sufficient for antirepressor activity

A collection of Tum mutants with premature stop codons arose in the point and insertion mutant screens. We also manually constructed a number of truncation mutants. Antirepressor activity reporter assays of these mutants were performed as before, but since the C-terminal gets truncated in the mutants, the calculated activities were normalised instead to the untagged *tum*4^-5^- control (see Figure 6A).

Tum activity matched the Tum$^-$ control for truncations terminating at or before residue 40. A low level of Tum activity was detectable for the Tum44V* mutant (the V is inserted after residue 44 in this insertion mutant), after which activity increases with increasing Tum length up to Tum73V. The full activity of Tum71V shows that Tum's antirepressor activity is wholly contained within the N-terminal domain. The super-activity of the Tum73V mutant indicates that, if anything, the DinI-like domain of Tum limits the antirepressor function of the NTD.

It was shown in an earlier study [12] that the translation product from ORF2 is inactive. We confirmed this result using an N-terminal 9 amino acid truncation of *tum*4^-5^- to the ORF2 start site (Tum$1^-4^-5^-$ in Figure 6A), but did not pursue any other N-terminal truncations.

It was originally thought that a minimal active Tum truncation would provide a useful starting point for structural studies. So with a view towards protein purification, a high-resolution series of targetted truncation mutants was made from an N-terminally His_6-tagged Tum template (*his-tum*4^-5^-). The complete set of these truncation mutants is summarised in Supplementary Figure S2, but the most notable ones were selected for a more detailed analysis shown in Figure 6B. More sensitive comparisons of

antirepressor activity can be drawn from the half maximal effective concentration (EC_{50}) values obtained by fitting Hill curves to the assay data.

The most active truncation of the N-terminally His_6-tagged Tum mutants was the His_6Tum51* construct, which showed even more activity than the His_6Tum72* mutant (the closest in length to the most active untagged truncation mutant). Quite in contrast with the untagged truncation mutants, the minimal active truncation, His_6Tum44*, was just as active as the His_6Tum control. This surprising result suggested that, in this instance, the His_6 tag cannot be considered independent of the target structure as it appears to assist in stabilising folding of this short truncation. Activity is nonetheless sharply lost over the next couple of residues: His_6Tum43* has an EC_{50} approximately twice that of His_6Tum44*, and His_6Tum42* showed no activity in our assay.

Directing structure prediction using the results of the mutagenesis studies

In spite of the high activity of some of the short truncation mutants, our attempts at purifying these Tum constructs for structural studies were unsuccessful. None of the Tum variants could be concentrated higher than 0.2 mg/mL before precipitating out of solution. So to make structural sense of the conserved residues of Tum and the various mutants, we turned to a structure prediction/modelling approach.

Given the two domain model for Tum and the independence of these domains, we pursued structural modelling on each domain separately. To begin with, the N- and C-terminal domains of Tum (residues 1–70 and 71–146) were submitted to the I-TASSER server [24], essentially a multi-template homology modelling approach. A single but high confidence candidate for the carboxy-terminal domain (CTD) (C-score of 1.27 and estimated RMSD from native of 1.1 ± 1.1) was returned, and primarily modelled on the solution structure for *E. coli* DinI (PDB accession 1GHH) [25]. As anticipated, the candidate structure showed significant overlap with the DinI template (see Supplementary Figure 4), but overlap in the C-terminal helix was especially good. A series of negatively charged residues in this helix in DinI have been implicated in mimicking single-stranded DNA [26]. Whilst overlap is good, two of the seven negatively charged residues appear to be absent in Tum.

For the NTD, the performance of I-TASSER was much poorer, with the five alternative candidates all giving low confidence scores. This was not unprecedented since none of the sequence homologs of the Tum NTD have a determined structure at present. Hence for the NTD, we turned to the Rosetta structure prediction suite [27], a more *ab initio* method based on physically-derived molecular energetics. A Monte Carlo search of the fold space for the global energy minimum produces of thousands of 'decoy' structures that map the energy landscape; good candidate structures are typified by deep energy wells. Since Tum truncations to 50/51 residues still displayed good activity, we proceeded by characterising this core fold. Decoys were generated for the first 51 residues of Tum, and a high-confidence consensus fold, depicted in Figure 7, was evidenced by obvious funnelling of the energy landscape towards the lowest energy decoy (see Supplementary Figure S5).

This consensus structure contains two extended α-helices packed side-by-side, but offset at a characteristic angle with respect to each other. A set of core hydrophobic residues pin the two helices in this orientation, and those making the largest contributions to the overall low score of this decoy are labelled in Figure 7A. The hydrogen bonds predicted using the Rosetta scoring function are also shown in the figure, and a particularly high-scoring contact was found between the aspartate at residue 30 and the peptide backbone in the loop between the helices.

All of the residues that made the biggest contributions to the low score of the predicted structures also displayed inactivating mutations in the point mutant screen; this both validated the predicted importance of those residues by Rosetta, but also assisted in functionally classing those point mutants apart from the others (see Figure 7C).

Discussion

Our bioinfomatics show that 186 Tum is representative of a wide family of antirepressors that is likely to be much larger than presently realised. This family includes previously identified antirepressor genes from the PSP3 and Fels-2 phages, the latter also known to be similarly LexA-sensitive via its *tum* homolog [15]. In those cases, the region of Tum homology is split into two independently translanted genes. Here we have shown that 186 Tum is a fusion of these gene products that can instead be split into two functional domains: an N-terminal domain that is exclusively responsible for the antirepressor activity and a C-terminal domain, which is already known to show strong homology with the *E. coli* DinI protein (DNA-damage inducible protein I) [14]. There is no obvious value in such a fusion or separation, but independent production of the DinI homolog is conserved due to internal translation start sites in the 186 *tum* gene.

The CTD appears to play no role in derepression of p_{R}; if anything, it reduces the activity of the NTD. This is further supported by the fact that removal of ORFs 4 and 5 had no impact on antirepressor activity. Its purpose would appear to be associated with that of the many other such DinI-like proteins found in bacteriophage [14]. These often belong to SOS operons produced from LexA-repressible promoters, revealing a tight integration of such proteins within the SOS response. Nonetheless, the purpose of these DinI-like factors is still unknown, and made more complicated by the fact that the exact role of DinI is still in question. The DinI protein is known to be involved in stabilising RecA filaments as part of the prokaryotic SOS response [28,29], and is, like 186 Tum, expressed from a LexA-repressible promoter [30]. Given that multiple binding modes have been reported for the *E. coli* DinI protein in regulating the formation of RecA-DNA filaments [31], it seems likely that some of these DinI-like proteins found in phage might favour one operating mode over another to manipulate the SOS response in favour of virion production. Indeed, the structure predicted for the Tum CTD showed especially good overlap with DinI over the most C-terminal helix, the negatively charged residues of which are implicated in mimicking single-stranded DNA [26]. Whilst the nearest matching residues in Tum do not replicate this string of negative charges quite as well, it seems likely that this domain mimics some function of DinI, and may even play an inhibitory role in 186 induction.

The antirepressor activity of Tum is associated with its NTD. This was most clearly evidenced by a number of Tum truncation mutants shorter than 70 amino acids that still retained full (or in some cases greater-than wild-type) antirepressor activity. Some activity was lost through truncation from 60 to 50 residues, removing the predicted β-strand region as a result, but most of the activity was lost between residues 50 to 46 as the truncation moved into the second predicted α-helix. The antirepressor activity of Tum is captured within these two α-helices, and comparison with the predicted structure showed that the helix boundaries closely matched those predicted directly from the sequence.

The predicted fold of the NTD is well-supported by antirepressor-specific point mutants. Early mutants of Tum were obtained by mutagenising the whole phage genome and screening for those that rendered phage 186 insensitive to SOS induction by DNA-damaging agents like mitomycin C or UV irradiation [10,11]. Here, to focus on just the antirepressor activity of Tum, we decoupled *tum* expression from the SOS response and designed an assay strain that reports only on relief of repression at p_{R}. All the point mutants obtained from this targetted screen were located within the first two α-helices of the protein, strong evidence for a small, independently-folding antirepressor domain. The restricted location of these mutants also overlapped well with the region of greatest sequence homology found for this domain. The point mutants were found centred about the loop between the helices, though no point mutations were found within the loop. Thus the co-location of these two clusters of mutants in the predicted structure via a 180° turn at the intervening loop was unsurprising. The co-locating hydrophobic point mutants also showed high sequence conservation, confirming their involvement in core hydrophobic packing of the domain. The insertion mutants found within the loop were only mildly-disruptive, further confirming the importance of the surrounding residues in holding the two helices together. The likely shape-changing T21P and A22P mutations provide evidence that straining the formation of the loop

is deleterious. The aspartate at residue 30 was particularly sensitive to moderate point mutations, the D30E mutation being a case in point, and further noted for its high sequence conservation. This residue may well be critical to folding and positioning of the helices, as evidenced by the high-scoring hydrogen bond it forms with the loop.

The fact that no point mutants were found after the first two α-helices, and that truncation of this region did not effect any substantial loss in activity, left the function of the residues between the end of the second α-helix and the start of the DinI-like domain in question. One possibility is that these residues provide (limited) structural support, and that the stringency of the screening method may have filtered out some mild but structurally-relevant mutations in that region. Indeed, the hydrophobic residues L35 and I38 were both exposed in the Tum51* decoy, but scored highly in the Tum72* decoy, with similar trends observed for the predicted structures of the homologs. Alternatively, these residues might represent an extended flexible linker, and the difficulty in obtaining a predicted structure for those residues supports that hypothesis. Nonetheless, the core Tum51* structure is well-supported by the available experimental data and forms the basis for an understanding of its mode of CI antirepression.

Antirepressors can act by numerous mechanisms, including by disruption of repressor multimerisation [32], by targeting the protein for degradation [33], by inducing non-functional protein conformations [34], or by acting as DNA mimics to mask the DNA-binding site of the repressor protein [35]. With so many available modes of repressor inactivation, the antirepressor-repressor paradigm becomes a likely platform for promiscuous binding and hence interaction between phage and phage-like elements; the extent of such cross-talk is only just being realised [8,9,36]. As for the mechanism of Tum, it was previously noted that mutants of CI resistant to Tum but still able to repress p_{R} were not able to be found [13]. This favoured a mode of operation at function-critical regions, like by disruption of multimerisation or binding at the DNA-binding region of CI. Given the highly cooperative nature of CI binding at p_{R} and the ultrasensitive antirepression curves, disruption of multimerisation previously seemed more likely [13]. Whilst this mechanism still cannot be excluded, here our preferred mechanism is for Tum operating as a DNA mimic. This was prompted by the overall negative charge of the antirepressor domain, the high abundance of negatively charged residues sensitive to point mutation (5 negatively charged residues were found mutated versus one positively charged residue), and a strong hydrophobic core, all common features of DNA mimics [37]. A number of the negatively charged residues were less well-conserved amongst the N-terminal homologs, suggesting a role in providing repressor specificity. The four identified sites of charge reversal (E12K, E15K, E32K & E40K) will be particularly interesting candidates for future complementation studies with CI. The close structural homology with the fourth domain of UvrB added extra support for a model of DNA mimicry, since there is some evidence that this domain acts to partially suppress the DNA-binding capacity of the three much larger catalytic domains of the protein [38]. The UvrB domain crystallised as a dimer with a binding interface across the loop remarkably similar to that of the predicted Tum dimer (the two are shown overlaid in Supplementary Figure S7). We have noted that this mode of dimerisation brings the highly-conserved cysteine residues in the Tum NTD into close proximity, possibly implicating a functional role for this contact.

The *in vivo* antirepressor activity measured using the assay strain displayed a highly ultrasensitive response, which matches well with previous *in vitro* results [13]. This sharp threshold occurred at levels of induction that were higher than anticipated; complete derepression was only observed when *tum* was cloned in a plasmid with the high-copy ColE1 origin [39], with the most sensitive region of the derepression curve occurring at levels of induction that are on par with the strength of the single-copy p_{R} promoter (see Figure 3). The implication is that in order to overcome lysogenic levels of CI, the p_{95} promoter in the single copy 186 prophage must be quite strong, or alternatively, that Tum's antirepressor activity may be enhanced by the SOS response.

Materials and Methods

Strains and media

All reporter strains were derived from the K-12 *E. coli* strain NK7049 (Δ*lac*χ74 *galOP*308 *rpsL su*$^-$) from R. Simons [40]. Propagation and cloning of plasmids with the *pir*-dependent R6Kγ origin (those derived from pAH144) was done in BW23473 [41], which is *pir-116* to permit higher copy numbers. XL1-Blue (Stratagene) was the host for point mutant screening.

Strains were grown in Luria-Bertani broth (LB) with the addition of antibiotics at the following concentrations: either 100 μg/mL carbenicillin or 100 μg/mL ampicillin (for maintaining pZE15-based plasmids), 50 μg/mL kanamycin (for maintaining pUHA-1), 5 μg/mL tetracycline (for maintaining the F′ plasmid in XL1-Blue), and 25 μg/mL spectinomycin (to select for colonies with an integrated pAH144 cassette; omitted during the assays). Strains were grown at 37°C, except where otherwise indicated.

Tum activity reporter strains

The *in vivo* activity of Tum was assayed indirectly by its ability to relieve repression at the p_R promoter by CI. The reporter phage pBC2 − HS − $F_L^+ \underline{p_R p_L} F_R^+$ from [20] was integrated at the λ attachment site of NK7049. This reporter contains both flanking CI binding sites for enhanced repression, and is oriented with p_R reading into *lacZ* (with a Δ*YA* deletion). The fragment truncates the *c*I gene and has an amber mutation in *apl*. Constant production of CI was from a constitutive P_{bla} promoter. Base pairs −171 to +9 of P_{bla} (chosen to match [42]) were polymerase chain reaction (PCR) amplified from pTL61T [43] with XbaI and KpnI tails and cloned into the CRIM plasmid pAH144 [44]. The 186 *c*I gene, including its wild-type ribosome binding site (RBS), was PCR amplified and cloned downstream in the EcoRI site to form pAH144-P_{bla}-*c*I. This CI expression module was integrated into the HK022 attachment site in NK7049; for *c*I$^-$ controls an empty pAH144 module was integrated. The final antirepressor activity reporter strain () was NK7049 (λRS45-pBC2 − HS − $F_L^+ \underline{p_R p_L} F_R^+$) (pAH144-$P_{bla}$-*c*I)$_{HK}$ ((P_{bla}-*c*I)$_{HK}$) pUHA-1, where pUHA-1 is a p15a origin plasmid with kanamycin resistance that supplies Lac repressor (obtained from H. Bujard).

Inducible levels of the various *tum* constructs were supplied by cloning each one downstream of P_{lac} in pZE15 [45]. pZE15-*tum*4$^-$5$^-$*his* was made by PCR amplifying the *tum*4$^-$5$^-$*his* template from pET-Tum His_6 [13], including the pET RBS and with a HindIII tail on the upstream primer, digesting with HindIII and partially digesting with BamHI (there is a BamHI site within the *tum* gene), and cloning into the same sites in pZE15. Plasmid pZE15-*tum*$^+$ *his* was made in the same way with a pET3a-*tum*$^+$ intermediate. The His_6 tags were removed from pZE15-*tum*4$^-$5$^-$*his* and pZE15-*tum*$^+$ *his* by PCR amplifying the *tum* templates without the His_6 tag and introducing a stop codon and AvrII site on the downstream primer tail. These were reintroduced into a pZE15-*tum*4$^-$5$^-$*his* backbone digested with NdeI and AvrII to replace the entire *tum*4$^-$5$^-$*his* sequence. The *tum* constructs in pZE15 were all transformed into ; empty pZE15 was used as a *tum*$^-$ control.

Generating point mutants of Tum

The screening strain (KES1033) was a 186(*c*I*tsp int*$^-$) lysogen of XL1-Blue [22], which provides LacZω from an F′ plasmid. Due to the temperature sensitive mutation in the CI repressor (*c*I*tsp*), when screening for Tum mutants, cells must be grown at the permissive temperature of 30°C. However this also provides an effective control to test for continued lethality of the lysogen, since growth at 41°C induces the prophage.

Plasmid pZE15-*tum*4$^-$5$^-$α was made by using a PCR-amplified *lacZ*α fragment (*lacZ* codons 3–42) from pMRR9 [20], with SacII and AvrII tails on the upstream and downstream primers respectively

and cloned into a SacII/AvrII digest of pZE15-*tum*4^-5^-*his*, thence replacing the histidine tag with the α-fragment.

Point mutations were generated from the pZE15-*tum*$4^-5^-\alpha$ template using error-prone PCR between primer 215 and the zalpha AvrII primer. The DNA was first amplified using the proof-reading Pfu polymerase (Stratagene). 1 μL of this reaction was then used as the template for 6 cycles of an error prone PCR with Taq DNA polymerase in the presence of 6 mM $MgCl_2$. The reaction was transformed into KES1033 and spread on plates with carbenicillin, tetracycline, 5-bromo-4-chloro-indolyl-β-D-galactopyranoside (X-gal) (40 μg/mL) and Isopropyl-β-D-1-thiogalactopyranoside (IPTG) (50 μM) producing colonies of varied size and blueness at 30°C. Plasmids from all mutants were isolated and transformed into for quantitative assays.

A number of the mutants incorporated errors at two different sites. Where these were not silent mutations or represented elsewhere, the mutations were separated by re-introducing each one into *tum*$4^-5^-\alpha$ using the QuikChange Site-Directed Mutagenesis Kit (Stratagene).

Generating insertion mutants of Tum

Linker insertion mutagenesis across the *tum*4^-5^-*his* gene was performed using the Genome Priming System-Linker Scanning (GPS-LS) system (New England BioLabs), which produces 5 codon insertions (6 versions). The insertion library was prepared in three steps: (1) the gel-extracted small fragment of a pZE15-*tum*4^-5^-*his* HindIII/BtgI digest was subject to GPS reaction with the Transprimer-5 donor plasmid (with kanamycin resistance), (2) the reaction was cleaned, religated into the HindIII/BtgI digested backbone and transformed into the screening strain with selection for ampicillin and kanamycin, (3) the resulting colonies were pooled for isolation of the plasmid DNA and removal of the kanamycin resistance cassette by digestion with PmeI and then recircularisation. This library was transformed into the screening strain and plated onto media containing carbenicillin and X-gal (30 μg/μL). The screening strain was NK7049 (λRS45pMRR9 – FspI – $cI^+\underline{p_R}p_Lapl_{\mathrm{am}} - lacZYA$), a λRS45 recombinant [40] of pMRR9 with a blunt ended FspI digest of phage 186($\overline{apl}_{\mathrm{am}}$) ligated into the SmaI site. The 186 fragment encompasses the switch region from the beginning of *int* to the beginning of *c*II and clones with p_R reading into *lacZ* were chosen for this assay. This construct provides both lysogen-like levels of CI and a p_R-*lacZ* reporter, giving blue/white screening for active/inactive Tum.

Clones from the library with a range of LacZ activities were selected, sequenced over the *tum*4^-5^-*his* gene and plasmids isolated for those with distinct insertion events. These plasmids were transformed into for quantitative assays. Note that a point mutation (F56L) was discovered in all of the non-truncating insertion mutants except Tum3(CLNNR) and Tum83(CLNKI), and in Tum74V* of the truncating insertion mutants, implying that an error arose during an early cycle of PCR amplification of the *tum*4^-5^-*his* template. This mutation had little impact on activity as judged by the insertion mutants containing this point mutation that matched wild-type activity and for this reason was disregarded.

Cloning the truncation mutants of Tum

Primers designed to add a stop codon and AvrII restriction site after residues 37, 46, 50 and 60 were used to amplify a *tum*4^-5^- template, and the products were cloned into an NdeI/AvrII digest of pZE15-*tum*$4^-5^-\alpha$. The N-terminally His_6-tagged *tum*4^-5^- truncation series was generated from a pZE15-*his*-*tum*4^-5^- template, prepared by PCR amplifying the N-terminal His_6 tag from pET15b (Novagen), with a HindIII site in the tail of the upstream primer (the same used for amplifying *tum*4^-5^-*his* from pET3a), and cloning into the pZE15-*tum*4^-5^- backbone digested with HindIII and NdeI. Truncation mutants were prepared from this template using primers designed to add a stop codon and BamHI site directly following residues 34, 39, 42, 44, 47, 49, 51, 55, 56, 57, 58, 61 and 62, and cloned back into the template backbone digested with NdeI and BamHI. Truncation mutants terminating at residues 63, 67, 68, 69, 70, 72 and 74 were similarly prepared by PCR amplification, but cloned first into pET15b between the NdeI

and XhoI sites, before being transferred into the pZE15-*his*-*tum*$4^{-}5^{-}$ template by digestion with NcoI and BamHI.

LacZ assays

LacZ assays were performed essentially as described [45] with some minor amendments. Briefly, Luria-Bertani broth with appropriate antibiotics was innoculated from freshly streaked colonies for overnight growth in 96-well plates with a sealed lid. The following day these were adjusted to an optical density at 600 nm (OD_{600}) of 0.3, sub-cultured 1 in 50 into equivalent broth, but with added IPTG to 0, 2, 5, 10, 15, 20, 25, 35, 50, 70, 100 and 200 µM (or a subset thereof), and grown to late log phase (OD_{600} $0.6 - 0.8$). Cells were chilled, then incubated in Lysis Buffer [45] for $15'$ at 30°C before addition of *o*-nitrophenyl-β-D galactose (ONPG) and assayed over time at 28°C for increasing absorption at 414 nm in a Multiskan Ascent plate reader. All data analysis was performed in R [46] with four-parameter log-logistic regression (Hill fits) calculated using the 'drc' package [47].

Structural modelling

Tum was modelled as two separate domains with domain boundaries chosen as described in the text. *Ab initio* modelling of the NTD was performed using Rosetta version 3.2.1, essentially as described [48]. Briefly, a library of 3- and 9-residue fragments was generated for each protein sequence using the Robetta server. These were supplied to the 'abrelax' protocol that follows the initial coarse-grained fragment-based search with an all-atom energy minimisation 'relax' step. The protocol was run with the optional flags as recommended in the Rosetta manual. 20000–30000 decoys were generated for each sequence and the 1000 lowest-scoring decoys submitted to the 'cluster' application, which clusters decoys into groups by RMSD. The cut-off was typically set to 2 Å, and additional decoys generated if no sizeable clusters were found. Funneling towards the lowest-energy structure and distinction between multiple local minima was made by plotting the score of the decoys versus their RMSD from the candidate decoy. Hydrogen bonds and per-residue score terms were identified with the aid of PyRosetta [49].

Acknowledgments

Rachel A. Schubert provided the assay strain and raw data for many of the Tum mutants. Linda Shearwin provided a number of the Tum truncation mutants. Adam C. Palmer and Monica O'Shea helped to generate and screen the insertion mutant library. Iain Murchland provided much useful discussion. Computer time was provided by eResearch SA. This work was supported by ARC grants DP0665185 and DP110100824.

References

1. Frost LS, Leplae R, Summers AO, Toussaint A (2005) Mobile genetic elements: the agents of open source evolution. Nat Rev Microbiol 3: 722-32.

2. Rosario K, Breitbart M (2011) Exploring the viral world through metagenomics. Curr Opin Virol 1: 289-97.

3. Canchaya C, Fournous G, Chibani-Chennoufi S, Dillmann ML, Brüssow H (2003) Phage as agents of lateral gene transfer. Curr Opin Microbiol 6: 417-24.

4. Quinones M, Kimsey HH, Waldor MK (2005) LexA cleavage is required for CTX prophage induction. Mol Cell 17: 291-300.

5. Wagner PL, Neely MN, Zhang X, Acheson DW, Waldor MK, et al. (2001) Role for a phage promoter in Shiga toxin 2 expression from a pathogenic *Escherichia coli* strain. J Bacteriol 183: 2081-5.

6. Cox MM (2007) Regulation of bacterial RecA protein function. Crit Rev Biochem Mol Biol 42: 41-63.

7. Kim B, Little JW (1993) LexA and λ Cl repressors as enzymes: specific cleavage in an intermolecular reaction. Cell 73: 1165-73.

8. Lemire S, Figueroa-Bossi N, Bossi L (2011) Bacteriophage crosstalk: coordination of prophage induction by trans-acting antirepressors. PLoS Genet 7: e1002149.

9. Tormo-Más MA, Mir I, Shrestha A, Tallent SM, Campoy S, et al. (2010) Moonlighting bacteriophage proteins derepress staphylococcal pathogenicity islands. Nature 465: 779-82.

10. Woods WH, Egan JB (1974) Prophage induction of noninducible coliphage 186. J Virol 14: 1349–1356.

11. Lamont I, Brumby AM, Egan JB (1989) UV induction of coliphage 186: Prophage induction as an SOS function. Proc Nat Acad Sci USA 86: 5492–5496.

12. Brumby AM, Lamont I, Dodd IB, Egan JB (1996) Defining the SOS operon of coliphage 186. Virology 219: 105–114.

13. Shearwin KE, Brumby AM, Egan JB (1998) The Tum protein of coliphage 186 is an antirepressor. J Biol Chem 273: 5708–5715.

14. Casjens SR, Gilcrease EB, Huang WM, Bunny KL, Pedulla ML, et al. (2004) The pKO2 linear plasmid prophage of *Klebsiella oxytoca*. J Bacteriol 186: 1818-32.

15. Bunny K, Liu J, Roth J (2002) Phenotypes of lexA mutations in Salmonella enterica: evidence for a lethal lexA null phenotype due to the Fels-2 prophage. J Bacteriol 184: 6235-49.

16. Nilsson AS, Haggård-Ljungquist E (2007) Evolution of P2-like phages and their impact on bacterial evolution. Res Microbiol 158: 311-7.

17. Pelludat C, Mirold S, Hardt WD (2003) The SopEΦ phage integrates into the *ssrA* gene of *Salmonella enterica* serovar Typhimurium A36 and is closely related to the Fels-2 prophage. J Bacteriol 185: 5182-91.

18. Altschul SF, Madden TL, Schäffer AA, Zhang J, Zhang Z, et al. (1997) Gapped BLAST and PSI-BLAST: a new generation of protein database search programs. Nucleic Acids Res 25: 3389-402.

19. Söding J, Remmert M, Biegert A, Lupas AN (2006) HHsenser: exhaustive transitive profile search using HMM-HMM comparison. Nucleic Acids Res 34: W374-8.

20. Dodd IB, Egan JB (2002) Action at a distance in CI repressor regulation of the bacteriophage 186 genetic switch. Mol Microbiol 45: 697–710.

21. Dodd IB, Shearwin KE, Sneppen K (2007) Modelling transcriptional interference and DNA looping in gene regulation. J Mol Biol 369: 1200–1213.

22. Pinkett HW, Shearwin KE, Stayrook S, Dodd IB, Burr T, et al. (2006) The structural basis of cooperative regulation at an alternate genetic switch. Molecular Cell 21: 605–615.

23. Wigley WC, Stidham RD, Smith NM, Hunt JF, Thomas PJ (2001) Protein solubility and folding monitored *in vivo* by structural complementation of a genetic marker protein. Nat Biotechnol 19: 131-6.

24. Roy A, Kucukural A, Zhang Y (2010) I-TASSER: a unified platform for automated protein structure and function prediction. Nat Protoc 5: 725–38.

25. Ramirez BE, Voloshin ON, Camerini-Otero RD, Bax A (2000) Solution structure of dini provides insight into its mode of reca inactivation. Protein Sci 9: 2161-9.

26. Voloshin ON, Ramirez BE, Bax A, Camerini-Otero RD (2001) A model for the abrogation of the SOS response by an SOS protein: a negatively charged helix in DinI mimics DNA in its interaction with RecA. Genes Dev 15: 415-27.

27. Das R, Baker D (2008) Macromolecular modeling with rosetta. Annu Rev Biochem 77: 363–382.

28. Lusetti SL, Voloshin ON, Inman RB, Camerini-Otero RD, Cox MM (2004) The DinI protein stabilizes RecA protein filaments. J Biol Chem 279: 30037-46.

29. Renzette N, Gumlaw N, Sandler SJ (2007) DinI and RecX modulate RecA-DNA structures in *Escherichia coli* K-12. Mol Microbiol 63: 103-15.

30. Yasuda T, Morimatsu K, Horii T, Nagata T, Ohmori H (1998) Inhibition of Escherichia coli RecA coprotease activities by DinI. EMBO J 17: 3207-16.

31. Galkin VE, Britt RL, Bane LB, Yu X, Cox MM, et al. (2011) Two modes of binding of DinI to RecA filament provide a new insight into the regulation of SOS response by DinI protein. J Mol Biol 408: 815-24.

32. Lewis RJ, Brannigan JA, Offen WA, Smith I, Wilkinson AJ (1998) An evolutionary link between sporulation and prophage induction in the structure of a repressor:anti-repressor complex. J Mol Biol 283: 907–912.

33. Bose B, Auchtung JM, Lee CA, Grossman AD (2008) A conserved anti-repressor controls horizontal gene transfer by proteolysis. Mol Microbiol 70: 570-82.

34. Wilke MS, Heller M, Creagh AL, Haynes CA, McIntosh LP, et al. (2008) The crystal structure of MexR from *Pseudomonas aeruginosa* in complex with its antirepressor ArmR. Proc Nat Acad Sci USA 105: 14832-7.

35. León E, Navarro-Avilés G, Santiveri CM, Flores-Flores C, Rico M, et al. (2010) A bacterial anti-repressor with SH3 domain topology mimics operator DNA in sequestering the repressor DNA recognition helix. Nucleic Acids Res 38: 5226-41.

36. Chai Y, Kolter R, Losick R (2009) Paralogous antirepressors acting on the master regulator for biofilm formation in *Bacillus subtilis*. Mol Microbiol 74: 876-87.

37. Putnam CD, Tainer JA (2005) Protein mimicry of DNA and pathway regulation. DNA Repair (Amst) 4: 1410–1420.

38. Wang H, DellaVecchia MJ, Skorvaga M, Croteau DL, Erie DA, et al. (2006) UvrB domain 4, an autoinhibitory gate for regulation of DNA binding and ATPase activity. J Biol Chem 281: 15227-37.

39. Lutz R, Bujard H (1997) Independent and tight regulation of transcriptional units in *Escherichia coli* via the LacR/O, the TetR/O and AraC/I_1-I_2 regulatory elements. Nucleic Acids Res 25: 1203–1210.

40. Simons RW, Houman F, Kleckner N (1987) Improved single and multicopy lac-based cloning vectors for protein and operon fusions. Gene 53: 85–96.

41. Lu F, Schumacher MA, Arvidson DN, Haldimann A, Wanner BL, et al. (1998) Structure-based redesign of corepressor specificity of the *Escherichia coli* purine repressor by substitution of residue 190. Biochemistry 37: 971-82.

42. Liang S, Bipatnath M, Xu Y, Chen S, Dennis P, et al. (1999) Activities of constitutive promoters in Escherichia coli. J Mol Biol 292: 19–37.

43. Linn T, St Pierre R (1990) Improved vector system for constructing transcriptional fusions that ensures independent translation of *lacZ*. J Bacteriol 172: 1077-84.

44. Haldimann A, Wanner BL (2001) Conditional-replication, integration, excision, and retrieval plasmid-host systems for gene structure-function studies of bacteria. J Bacteriol 183: 6384–6393.

45. Dodd IB, Perkins AJ, Tsemitsidis D, Egan JB (2001) Octamerization of λ CI repressor is needed for effective repression of P_{RM} and efficient switching from lysogeny. Genes Dev 15: 3013–3022.

46. R Development Core Team (2012) R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria. URL `http://www.R-project.org/`. ISBN 3-900051-07-0, `http://www.R-project.org/`.

47. Ritz C, Streibig JC (2005) Bioassay Analysis using R. Journal of Statistical Software 12.

48. Raman S, Vernon R, Thompson J, Tyka M, Sadreyev R, et al. (2009) Structure prediction for CASP8 with all-atom refinement using Rosetta. Proteins 77 Suppl 9: 89-99.

49. Chaudhury S, Lyskov S, Gray JJ (2010) PyRosetta: a script-based interface for implementing molecular modeling algorithms using Rosetta. Bioinformatics 26: 689-91.

50. Waterhouse AM, Procter JB, Martin DMA, Clamp M, Barton GJ (2009) Jalview version 2–a multiple sequence alignment editor and analysis workbench. Bioinformatics 25: 1189-91.

51. Krissinel E, Henrick K (2004) Secondary-structure matching (SSM), a new tool for fast protein structure alignment in three dimensions. Acta Crystallogr D Biol Crystallogr 60: 2256-68.

52. Sohi M, Alexandrovich A, Moolenaar G, Visse R, Goosen N, et al. (2000) Crystal structure of *Escherichia coli* UvrB C-terminal domain, and a model for UvrB-UvrC interaction. FEBS Lett 465: 161-4.

53. Wang C, Bradley P, Baker D (2007) Protein-protein docking with backbone flexibility. J Mol Biol 373: 503–19.

54. Palmer AC, Ahlgren-Berg A, Egan JB, Dodd IB, Shearwin KE (2009) Potent transcriptional interference by pausing of RNA polymerases over a downstream promoter. Mol Cell 34: 545–55.

Figure Legends

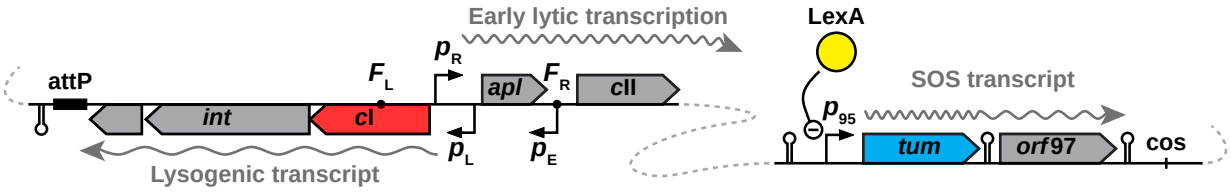


Figure 1. Genome organisation of bacteriophage 186 and the SOS operon. The CI repressor maintains lysogeny by repressing the p_R promoter and indirectly activates its own production by limiting transcriptional interference at p_L; the flanking CI binding sites, F_L and F_R, further enhance repression at p_R. Under normal conditions, the host LexA repressor keeps the p_{95} promoter of the prophage silent, but upon DNA damage, autocatalytic cleavage of the LexA repressor is stimulated, resulting in expression of host SOS genes but also the antirepressor Tum. Tum reversibly sequesters the CI repressor, and hence induces lytic development, during which the excisionase *apl* with the help of the integrase *int*, excises the phage genome for replication and packaging into phage particles. Upon infection of a host, lysogeny is established by CII which activates the p_E promoter causing integration of the phage via the attachment site attP. In this and the following figures, promoters are depicted by right angled arrows, genes by directed boxes and transcriptional terminators by stem loops. Wavy lines indicate the major transcripts; the frequency is indicative of the basal production rate.

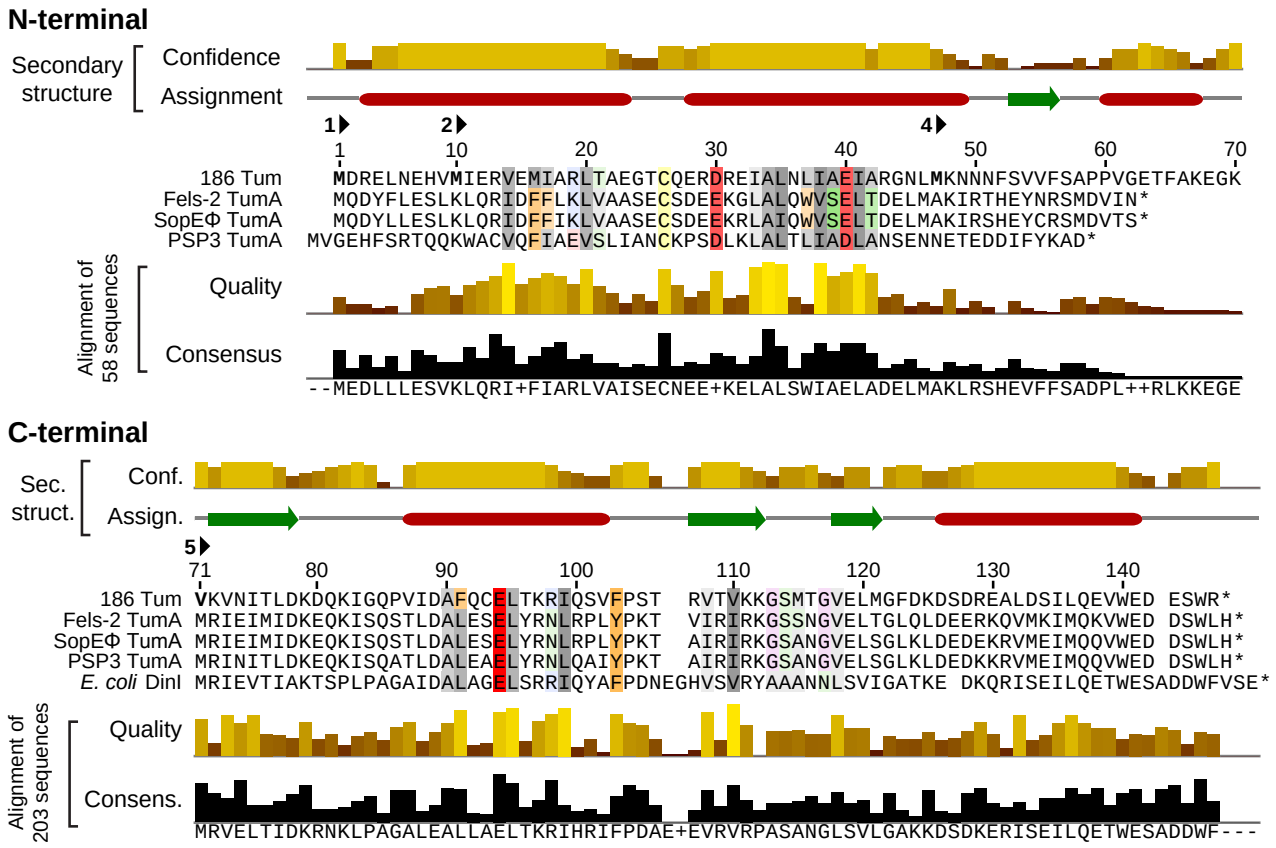


Figure 2. Sequence homologs of the Tum antirepressor. The N-terminal region of Tum is shown aligned with homologs identified in the literature (see the text). Unlike Tum, these presumptive antirepressors are split into two genes by an intervening stop codon. Shown aligned are the upstream genes from these phage. The downstream genes and C-terminal region of Tum are homologs within a superfamily of DinI-like proteins, so the alignment with the well-studied *E. coli* DinI protein is shown. These alignments were each derived from a master alignment of distant homologues identified using the HHsenser algorithm [19], which also produces secondary structure predictions as shown (green arrows for β-strand; red rods for α-helix). The alignment quality is that for each master alignment and is a summed likelihood for all mutation pairs at that position according to the BLOSUM62 matrix. Residues conserved throughout the master alignments are shown coloured according to residue type. The *tum* gene is translated from four alternative open reading frames (ORFs), which are marked by black triangles; ORF5 emphasises the independence of the C-terminal region. This figure was produced with the aid of JalView [50].

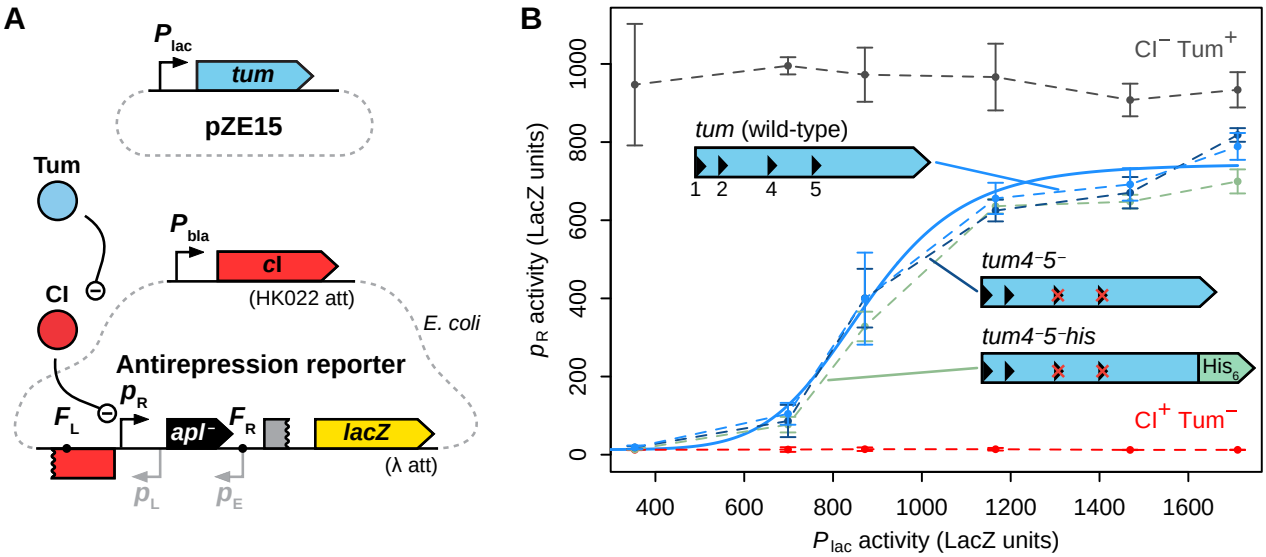


Figure 3. Relief of p_R repression by CI due to the antirepressor Tum is highly ultrasensitive *in vivo*. A. Antirepressor activity is measured *in vivo* by the extent to which an IPTG-inducible level of Tum can alleviate repression of the p_R promoter by CI. A consistent steady-state level of CI is supplied from a single-copy P_{bla} promoter. This pool of CI is sufficient to keep a single-copy p_R-*lacZ* reporter construct silent. *tum* constructs are cloned downstream of the P_{lac} promoter in the high-copy pZE15 vector. B. Increasing levels of wild-type Tum reduce the available pool of active CI and increase expression from the p_R promoter. The points joined by dashed lines are the mean results of LacZ assays of p_R activity over a range of induction levels of various Tum constructs; error bars represent 95% confidence limits. In the absence of Tum (red line; empty pZE15 vector) p_R stays repressed at all levels of induction. In the absence of CI (grey line; empty *c*I expression cassette), expression from p_R stays high. Removal of translation from ORFs 4 and 5, or addition of a C-terminal His_6 tag makes little difference to Tum activity. Since the nonlinear response of P_{lac} activity to inducer concentration can bias Hill fits, the level of Tum induction has been measured for each IPTG concentration as the equivalent LacZ activity produced from a P_{lac}-*lacZ* construct in pZE15 (see Supplementary Figure S1 for details). The solid line is a Hill fit of the wild-type data.

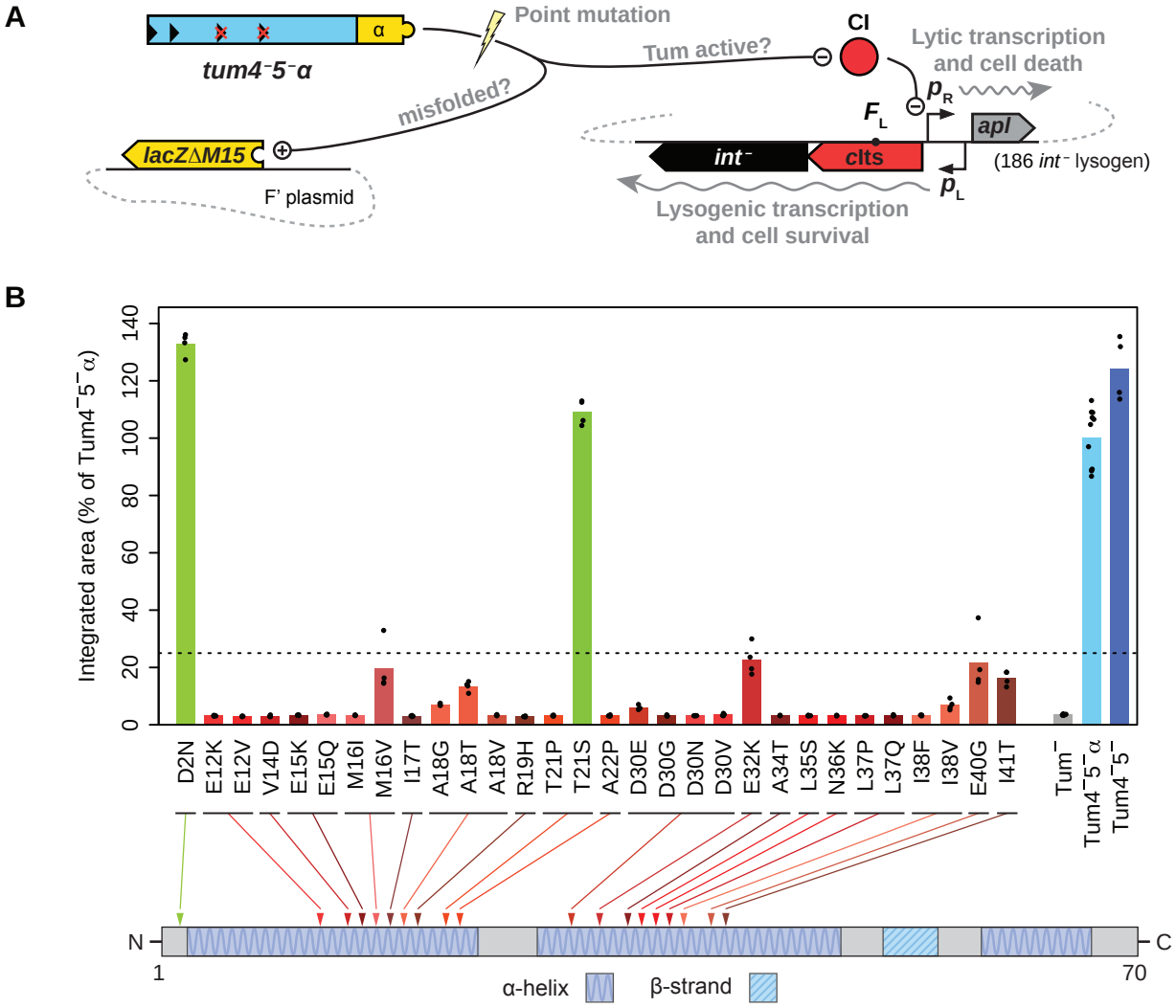


Figure 4. The antirepressor activity of Tum is located in the N-terminal domain. A. A library of point mutations across the entire *tum*4⁻5⁻α gene was generated by error-prone PCR and screened using a Prophage of Death (POD) strain in which mutations retaining antirepressor activity would successfully induce an *int*⁻ prophage. Induction of the *int*⁻ prophage, being unable to excise from the chromosome, results in cell death. Only mutants that are inactivating for antirepressor activity survive, and these can be further filtered for mutants causing minimal disruptions to folding using the C-terminal α-fragment fusion: successful complementation with the ω-fragment present in XL1-Blue strain produces a functional LacZ enzyme and produces blue colonies on plates with X-gal. This screening step allows for selection against mutants that result in misfolding, frame shifts or truncations. B. The resulting mutants were all contained within the first 41 residues. Activity curves for all the mutants were assayed as described in Figure 3. A summary value for each assay was calculated from the area under each curve using basic trapezoidal numerical integration between the (necessarily consistent) minimum and maximum induction levels. Mutants with reduced activity (high EC_{50}) give smaller areas, since the inflection point shifts closer to the maximum induction level. This summed area is displayed as a percentage of the mean area calculated for the fully-active control (*tum*4⁻5⁻α). Each point is derived from the curve measured for a single colony; most mutants were assayed on four independent repeats. A nominal 25% cutoff (horizontal dotted line) distinguishes active from inactive mutations. Note that both D2N and T21S were originally isolated as double mutants with R19H and D30G respectively. The location of each mutation is further illustrated by triangles that mark a map (to scale) of the N-terminal domain of *tum*. For reference, the predicted secondary structure (see Figure 2) is also shown.

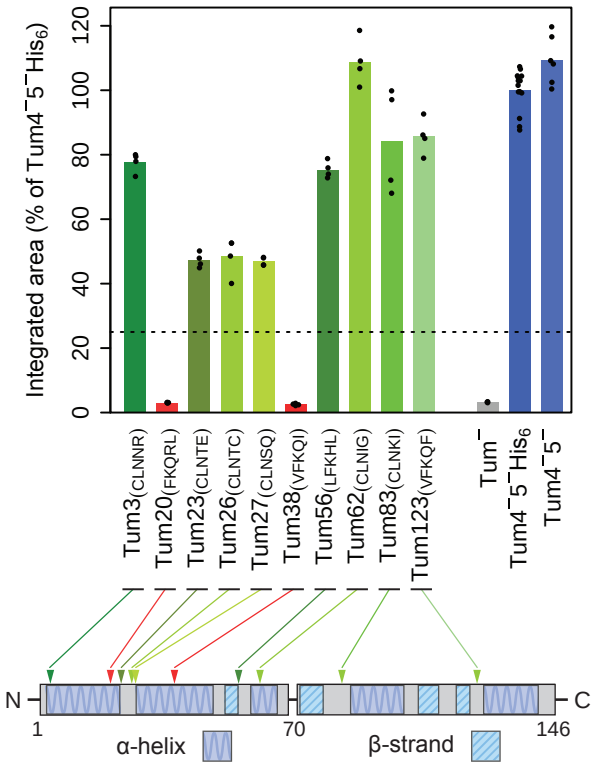


Figure 5. All regions of Tum but the first two α-helices are tolerant of amino acid insertions. Five amino acid insertions were introduced into an inducible *tum*4^{-}5^{-}*his* gene using a transposon mutagenesis approach as described in the text. They are labelled here by the sequence of residues inserted following each indicated position. The insertion mutants were assayed as for the point mutants (see Figure 4), each point representing the summed area under an activity assay curve for a given replicate. Here the area measurement is normalised to the mean area of the *tum*4^{-}5^{-}*his* control. A nominal 25% cutoff (horizontal dotted line) distinguishes active from inactive mutations. The location of each mutation is further illustrated by triangles that mark a scale map of the full-length *tum* gene. For reference, the predicted secondary structure (see Figure 2) is also shown.

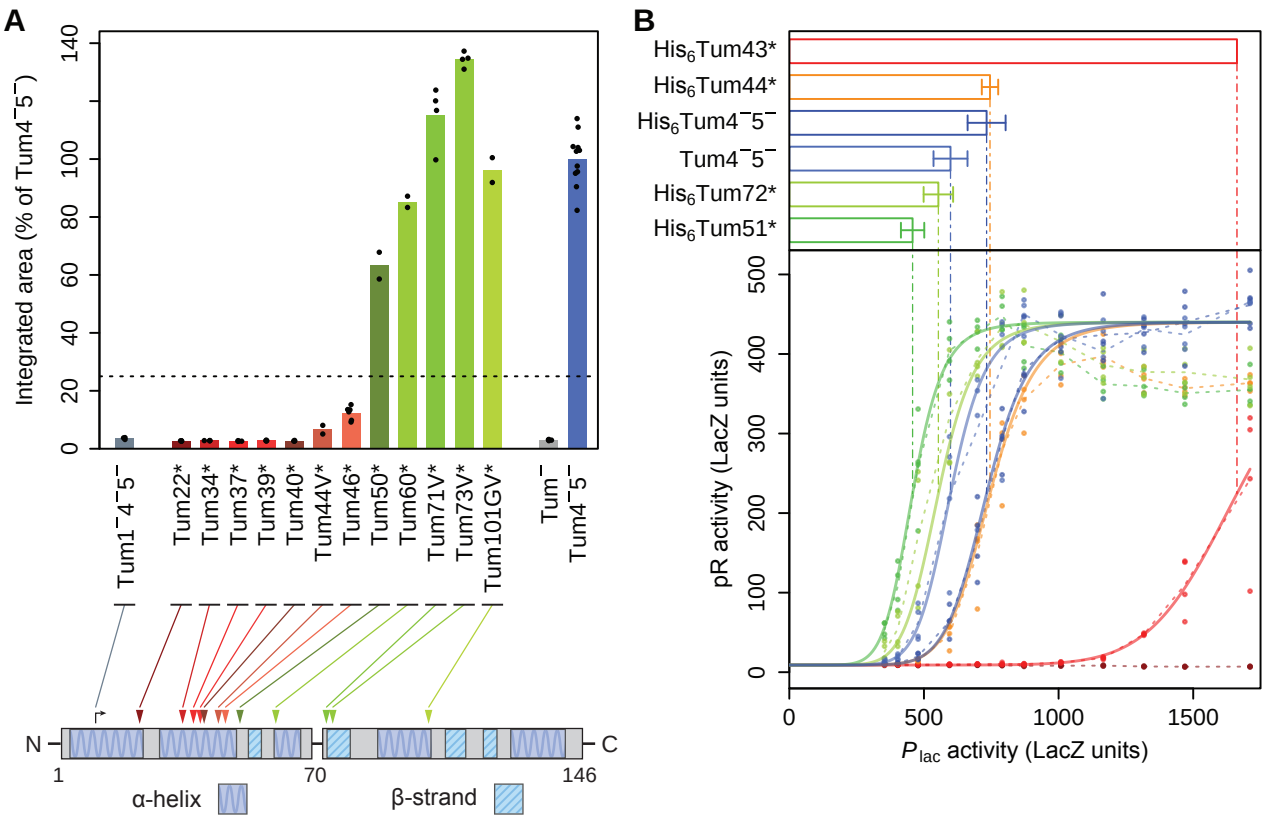


Figure 6. The Tum protein can be truncated to 44 residues and still retain function. A. Truncation mutants arising in the point and insertion mutant screens and a selection of targetted truncation mutants all based on the *tum*4^{-}5^{-} template were assayed in the antirepressor reporter strain. They are labelled here according to the number of residues matching the wild-type sequence; those from the insertion mutant screen only arose mid-insertion, so the non-native residue(s) are appended. Tum1^{-}4^{-}5^{-} is the ORF2 product, an N-terminal truncation. The area under each assay curve was calculated as for the point mutants (see Figure 4), and each replicate was normalised to the mean area of the *tum*4^{-}5^{-} control. A nominal 25% cutoff (horizontal dotted line) distinguishes active from inactive mutations. Locations of the terminating residues are further illustrated by triangles that mark a scale map of the full-length *tum* gene. For reference, the predicted secondary structure (see Figure 2) is also shown. B. A systematic series of C-terminal truncation mutants were prepared from an N-terminal *his-tum*4^{-}5^{-} fusion template and assayed in the antirepressor reporter strain. A selection of these curves measured for the most notable truncation mutants are shown in the lower panel; dotted lines follow the means of the data points. All but the flat response curve of His$_6$Tum42* (dark red) were fit with Hill curves (solid lines) using robust regression and fixed basal and maximum parameters. The mean activity of the Tum^{-} control was used for basal activity and the maximum parameter was set to that obtained for the *tum*4^{-}5^{-} control. In the panel above the curves are plotted the fitted half-maximal effective concentrations (EC$_{50}$s) for each fit; dot-dash lines match each bar graph to the respective assay curve. Error bars represent the standard error in the fit.

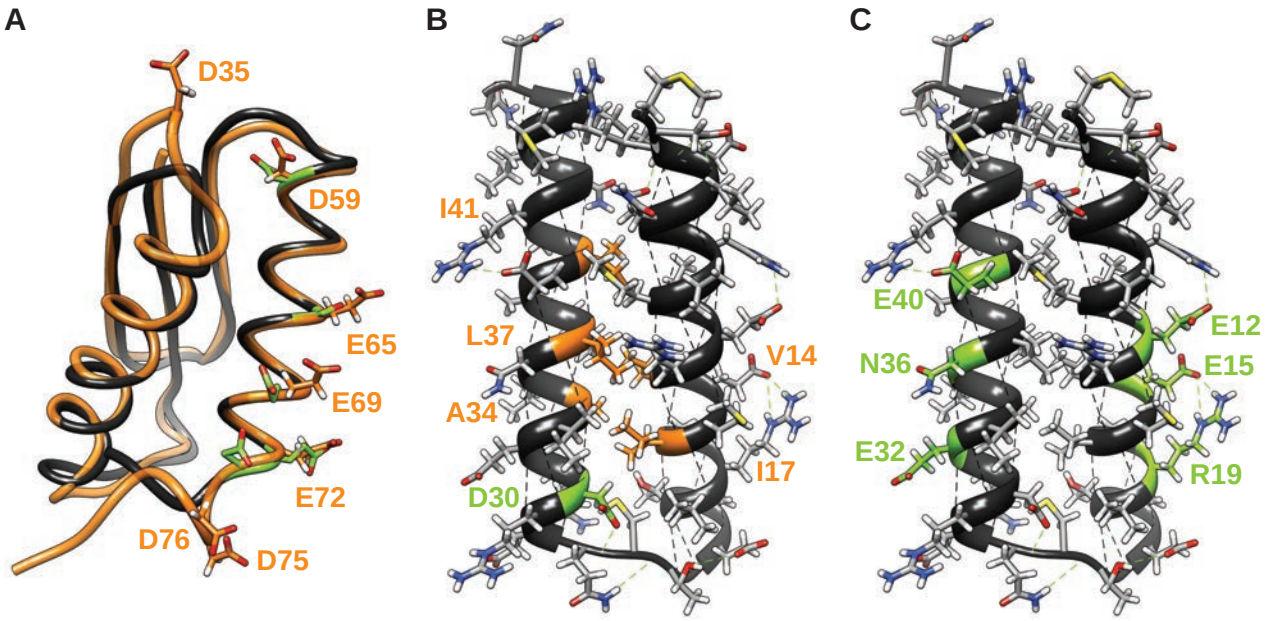


Figure 7. The set of Tum mutants is complemented by and assists structure prediction efforts. B. and C. *De novo* structure prediction of the first 51 residues of Tum using Rosetta [48] produced a well-defined fold that fixes the two extended α-helices relative to each other. Dashed lines indicate the hydrogen bonds predicted by Rosetta. Residues identified in the Rosetta scoring function as particularly important to folding are labelled in B. Hydrophilic residues found mutated in the point mutant screen are labelled in C.

Tables

Supporting Information

Supplementary Method S1

Structural homologs of this candidate structure were retrieved using the Protein structure comparison service Fold (PDBeFold) at the European Bioinformatics Institute (EBI) [51]. Around 120 distinct structural homologs were identified, with the best matches as judged by the root-mean-square deviation (RMSD) being synthetic four-helix bundle proteins. Given the small size of the fold, it is likely that many of these are false positive hits. However, a match with the NMR structure of the fourth domain of UvrB, stood out as being derived from *E. coli* and involved in the SOS response as part of the nucleotide excision repair pathway [52].

With a well-defined core 51-residue structure, we turned to exploring the likely conformations of the remaining residues in the NTD, using this core structure as a starting point. Truncation mutants to 72 residues produced high activities for both His_6-tagged and untagged variants, so decoys were generated from a 72-residue sequence with residues 12 to 41 held fixed to the structure of the lowest-scoring 51-residue decoy to reduce the search space. Only moderate funnelling towards the lowest-scoring decoy was observed, possibly indicating that these residues are not as structurally constrained. Decoys were also generated for a selection of the Tum sequence homologs identified in the HHsenser search and the best candidate decoys from these homologs are shown in Supplementary Figure S4 along with the best 72-residue decoy. The homologs were chosen specifically for their divergent homologies, but in spite of the obvious variations, these structures do provide some validation for the Tum72* decoy. The structure of the identified core functional region was similar amongst all the homologs, with a number of the well-conserved residues showing similar positioning. The homolog from Pantoea (Supplementary Figure S4D) showed particularly good clustering, and provides validation for the positioning of the putative β-strand in the Tum72* structure.

We previously found that Tum is predominantly dimeric at a concentration of 9.5 μM [13], and also that the ORF5 product is monomeric (unpublished data). From this we inferred that dimerisation is likely to be a function of the NTD, and explored this possibility using the Tum51* structure as the input for the Rosetta protein-protein docking protocol [53]. The lowest-scoring decoy produced an unusual elongated structure with the dimerisation interface located at the loop between the α-helices, as shown in Supplementary Figure S5.

Supplementary Table S8

Primers used in the course of this paper.

Amplifying the *c*I gene with wild-type RBS and EcoRI ends:

- ggaattcTGAATAGGTTTTATCG (108),
- ggtgaattcTCATTAGTTAACCTCGCT (295)

Amplifying the P_{bla} promoter from base pairs -171 to +9 (chosen according to [42, 54]) with XbaI and KpnI ends:

- tatctagaCAATTCTTGAAGACGAAAGGG (684),
- taggtaccGCATTTATCAGGGTTATTGTCTC (685); the template was pTL61T [43].

Amplifying *tum*4^-5^-*his*/*tum*$^+$ *his* genes from pET3a using 'B-PET HindIII' (164) and 'T7-rev/terminator' (315) primers; digested with HindIII and BamHI to clone into pZE15:

- cggaagcttaagTTTAACTTTAAGAAGGAGA (164),
- GCTAGTTATTGCTCAGCGGTGG (315)

Remove pET3a-derived His_6 tags using primers 215 (T7) and 686 (below), digesting with NdeI and AvrII to replace *tum*4^-5^-*his* in the pZE15-*tum*4^-5^-*his* vector: tacctaggTTAACGCCAGCTCTCATCTTCCCAC (686)

Rachel Schubert's (RS) zalpha AvrII and zalpha SacII primers: add downstream *lacZ*α fusion to *tum*4^-5^- by cloning into pZE15-*tum*4^-5^-*his* cut with SacII/AvrII, replacing His_6 tag starting from inside the thrombin cleavage tag. (NB: removes rgnB terminator by using the AvrII site)

- ttcc**ccgcgg**ATGATTACGGATTCACTGG (Upstream *lacZ*α SacII)
- gcg**cctagg**ttaTTCGCTATTACGCCAGCT (Downstream *lacZ*α AvrII)

Generate *tum*4^-5^- truncation mutants using the following primers, which append a stop codon and AvrII site, in combination with primer 215 for cloning into NdeI/AvrII digested pZE15-*tum*$4^-5^-\alpha$:

- Tum37: atcctaggttaTAGATTCAATGCGATTTCACG (680)
- Tum46: atcctaggttaTAGGTTGCCGCGTGCTATTTC (681)
- Tum50: atcctaggttaATTATTTTTCATTAGGTTGCCGC (682)
- Tum60: atcctaggttaAGGCGGTGCGGAAAAAACAAC (683)

To generate the N-terminal truncation, pZE15-*tum*$1^-4^-5^-$, use the following primer with the *tum*4^-5^- primer (686) and clone into NdeI/AvrII digest of pZE15-*tum*$4^-5^-\alpha$:

- 687 Tum2 5' NdeI: tggtggcatatgATTGAGCGGGTCGAAATG

Supplementary Figures

Julian Pietsch
Supervisors: Ian Dodd & Keith Shearwin
School of Molecular & Biomedical Science, The University of Adelaide, Australia

14th February, 2013

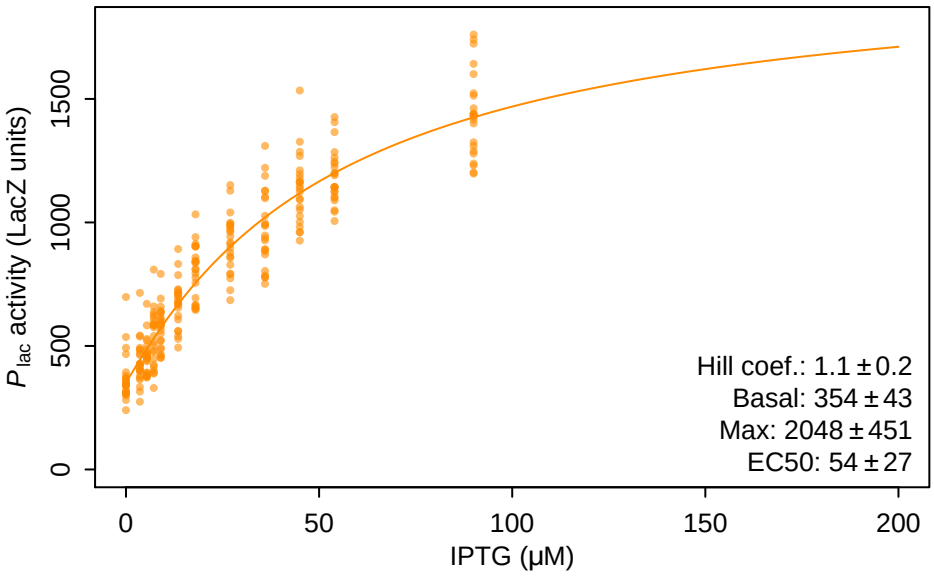


Figure 1: Measuring the production rate from P_{lac} in pZE15 as a function of inducer concentration. A *lacZ*O2^{-} gene (which includes a silent mutation to inactivate the internal *lac* repressor binding site and is Δ*YA*) with an upstream RNaseIII site [Linn and St Pierre, 1990] was cloned downstream of P_{lac} in the pZE15 plasmid, transformed into NK7049 with pUHA-1, and the β-gal activity measured at selected concentrations of IPTG [E. Cutts and K. E. Shearwin, manuscript in preparation]. A Hill curve (or log-logistic) fit was used to extrapolate to the IPTG concentrations used in this work and the fit parameters (Hill coef.: Hill coefficient; Max: maximum; EC50: half-maximal effective concentration) are indicated on the plot with 95% confidence limits. For pZE15, very little induction ultrasensitivity was observed compared with previous observations for pZS45 [Palmer et al., 2009], though given the high basal production rate it would appear that the curve lies well outside the normal region of ultrasensitivity in any case. The high basal production rate is not unexpected for such a high-copy plasmid (around 50–70 copies per cell [Lutz and Bujard, 1997]) when compared with the basal P_{lac}-*lacZ* production rates we have observed for comparable single-copy plasmids.

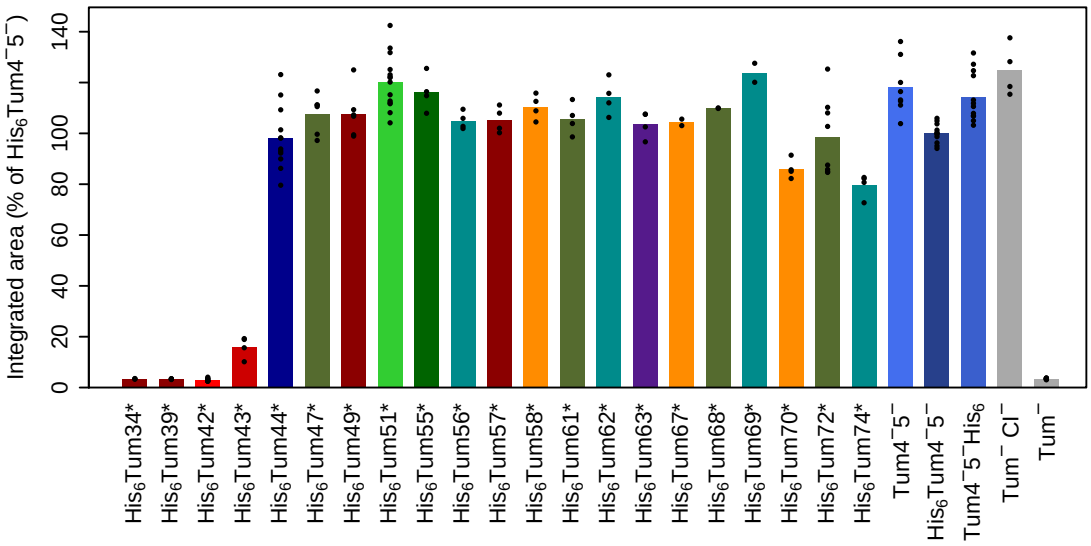


Figure 2: With an N-terminal His-tag, Tum can be truncated to 44 amino acids without losing its antirepressor activity. A series of N-terminally His-tagged Tum truncation constructs were cloned into pZE15 and assayed in the antirepressor assay strain. The area under each Tum assay curve was calculated as described in the main text and is shown normalised to the value obtained for the His_6Tum control measured on the same day. The unexpectedly low area seen for His_6Tum70*, His_6Tum72* and His_6Tum74* is partially a result of the observed suppression of antirepressor function as induction of the mutant becomes saturated.

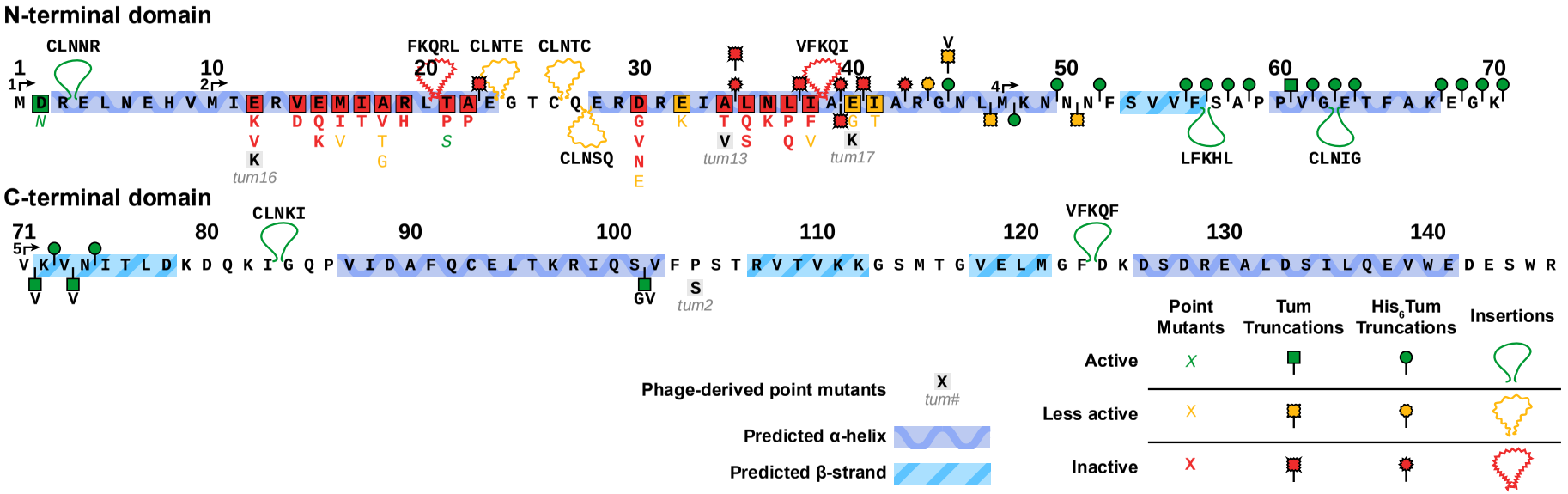


Figure 3: The full sequence of Tum annotated with all the mutations found in the paper. 'Inactive' mutations were those that showed no difference from the activity of a Tum$^-$ control, 'Less active' mutations were those that showed less activity than the wild-type control but more than the Tum$^-$ control, and 'Active' mutations were those that showed equal to or greater activity than the relevant wild-type control.

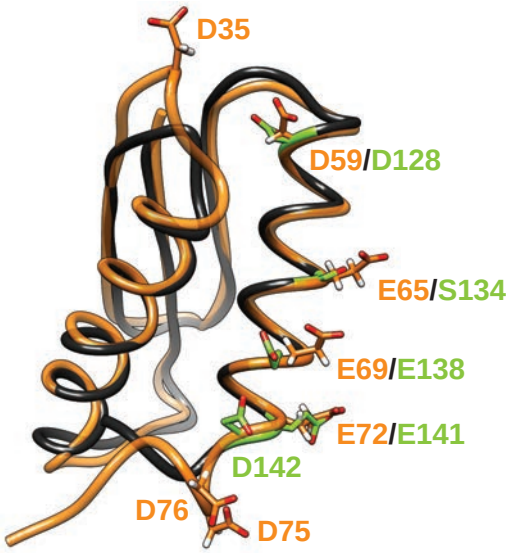


Figure 4: A homology model of the *tum* ORF5 product — the CTD of Tum— shows that it overlaps well with the experimental *E. coli* DinI structure. The structural model of the Tum CTD obtained using I-TASSER Roy et al. [2010] is shown in black, and the closest matching DinI conformer from the NMR ensemble (PDB ID: 1GHH Ramirez et al. [2000]) is shown overlaid in orange. The side chains of the seven negatively charged residues thought to be involved in mimicking single-stranded DNA Voloshin et al. [2001]; Casjens et al. [2004] are displayed on the DinI structure in orange and the corresponding best matches from the Tum CTD are shown highlighted in green.

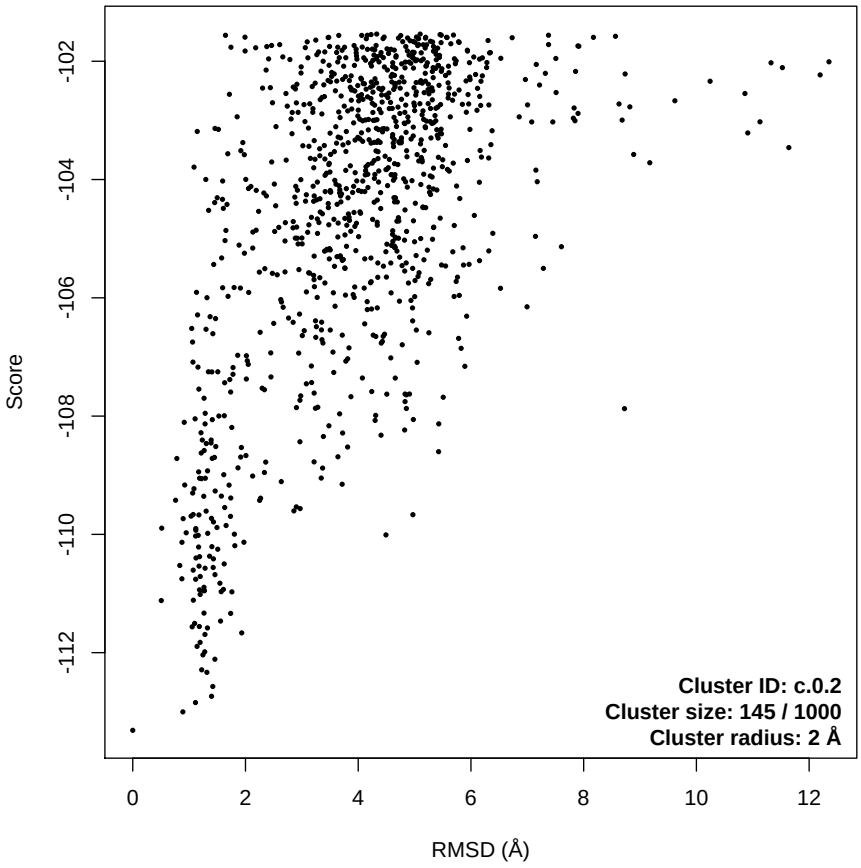


Figure 5: The validity of the predicted Tum51* structure is confirmed by funnelling of the energy landscape. The Rosetta score for each of the top 1000 Tum51* decoys is plotted versus the RMSD of that decoy from the lowest scoring decoy.

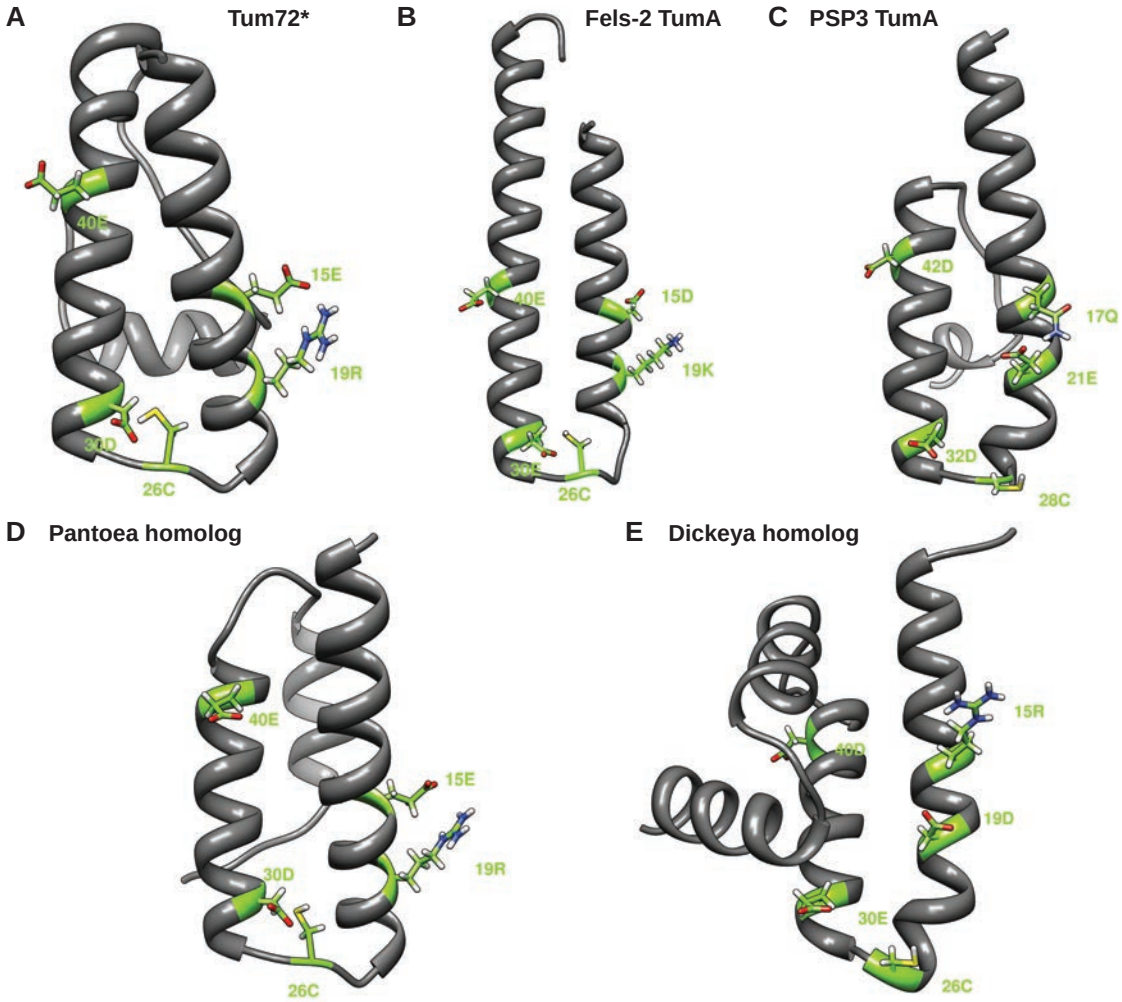


Figure 6: The ubiquitous presence of two helices separated by a loop in Tum and its homologs validates that core structure. Shown are the best scoring decoys predicted for some selected homologs of the 186 Tum N-terminal domain (NTD). A. the Tum72* truncation (NTD of 186 Tum). B. Fels-2 TumA (NCBI accession: YP_001718752). C. PSP3 TumA (NCBI accession: NP_958094). D. a Tum homolog found in Pantoea (NCBI accession: ZP_09512655). E. a hypothetical protein from Dickeya (NCBI accession: YP_002987696). A, B, and C are confirmed antirepressors; D and E are putative antirepressors. The Pantoea homolog gave the best clustering of low-scoring decoys. The Dickeya homolog was the most distant of those chosen. Highlighted in green are some of the best-conserved hydrophilic residues (see Figure 2 in the main text) and also the highly conserved cysteine in the loop between the two primary α-helices.

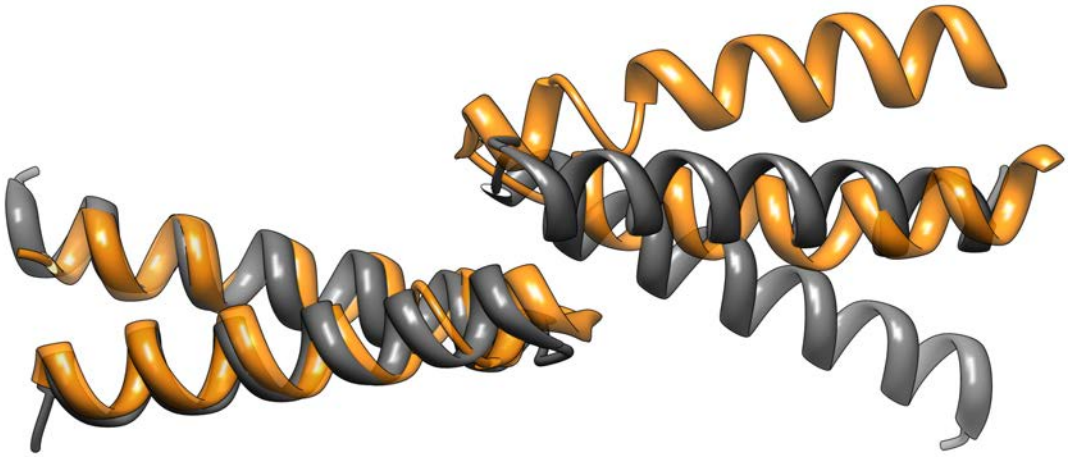


Figure 7: A close structural homolog of the predicted structure for Tum51* was the fourth domain of UvrB (PDB ID: 1QOJ) which crystallised as a dimer via the loop. This structure, depicted here in orange, is shown overlaid on the best candidate dimer obtained using the Rosetta docking protocol (run essentially as described [Wang et al., 2007]) for two monomers of the Tum51* structure (in grey).

References

Casjens, S. R., Gilcrease, E. B., Huang, W. M., Bunny, K. L., Pedulla, M. L., Ford, M. E., Houtz, J. M., Hatfull, G. F., and Hendrix, R. W. (2004). The pKO2 linear plasmid prophage of *Klebsiella oxytoca*. *J. Bacteriol.*, 186(6):1818–32.

Linn, T. and St Pierre, R. (1990). Improved vector system for constructing transcriptional fusions that ensures independent translation of *lacZ*. *J. Bacteriol.*, 172(2):1077–84.

Lutz, R. and Bujard, H. (1997). Independent and tight regulation of transcriptional units in *Escherichia coli* via the LacR/O, the TetR/O and AraC/I_1-I_2 regulatory elements. *Nucleic Acids Res.*, 25(6):1203–1210.

Palmer, A. C., Ahlgren-Berg, A., Egan, J. B., Dodd, I. B., and Shearwin, K. E. (2009). Potent transcriptional interference by pausing of RNA polymerases over a downstream promoter. *Mol. Cell*, 34(5):545–55.

Ramirez, B. E., Voloshin, O. N., Camerini-Otero, R. D., and Bax, A. (2000). Solution structure of dini provides insight into its mode of reca inactivation. *Protein Sci.*, 9(11):2161–9.

Roy, A., Kucukural, A., and Zhang, Y. (2010). I-TASSER: a unified platform for automated protein structure and function prediction. *Nat. Protoc.*, 5(4):725–38.

Voloshin, O. N., Ramirez, B. E., Bax, A., and Camerini-Otero, R. D. (2001). A model for the abrogation of the SOS response by an SOS protein: a negatively charged helix in DinI mimics DNA in its interaction with RecA. *Genes Dev.*, 15(4):415–27.

Wang, C., Bradley, P., and Baker, D. (2007). Protein-protein docking with backbone flexibility. *J. Mol. Biol.*, 373(2):503–19.

A

Fitting growth curves

During growth to log phase in 96-well plates, the A_{620}s of growing cultures can be recorded using a Multiskan Ascent plate reader at a number of time points during the course of any given assay. Such data can then be used to determine a growth rate for each strain. Numerous methods for estimating the 'ideal' (exponential) growth rate were trialled, including linear regression of the log-transformed OD_{600}s, weighted non-linear regression with an exponential model on the untransformed OD_{600}s, regression using generalised linear models (GLMs) with alternative link functions, and nonlinear regression using a Gompertz model on the log-transformed OD_{600}s.

Before considering the different models, general improvements can be made by employing some preliminary formatting of the data. It was found that substantial statistical improvements could be made by taking a per-well offset using the A_{620} reading of the plate at time $t = 0$. This relies on the fact that the signal due to the cells is hidden relative to experimental noise at that time point, and also that the systematic error between wells (due to, e.g., small differences in volume or the background absorbance of the plate) is sufficient to warrant such a correction. By comparing the absorbances measured for the first time point with the absorbances measured at another time point early enough that

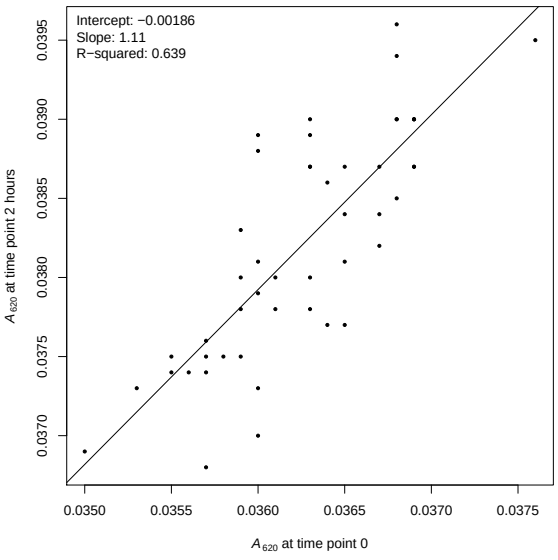


Figure A.1: Well-to-well variations in 96-well plates are correlated over time. The variation in A_{620} per well is a systematic error that persists over time. This is demonstrated in a growing culture where in the first two hours, the measurements (both with OD_{600} close to zero) are well-correlated. Hence growth curves may take the reading at time zero as the background reading for the extent of the assay.

the cells are still not evident above background noise, it can be seen that the readings are correlated (see Figure A.1), which supports the relevance of such a systematic error. To correct for this error, the reading at $t = 0$ is used as a per-well background measurement to offset the readings from all subsequent measurements made over the duration of the assay; this first data point is also then discarded.

A.1 Log-linear fits

Here we start by running some trial fits of growth data for E4300 (pIT-SL)$_{\lambda}$ (pIT-CH-*pR*-*gfp*)$_{\mathrm{HK}}$ pUHA-1 pZC320t-*lacZ* pMTS-*pR*-*tum*$^{+}$ (glycerol Julian Pietsch (JP)#157.1) in LB with 50 μg/mL Kanamycin (Kan), 30 μg/mL Ampicillin (Amp) and 4 μg/mL Tet. An upper threshold on exponential growth was derived by aiming for some desired maximum fold-reduction in the residual of the n^{th} point, using a fit of the first $n - 1$ points. Both a log-linear GLM assuming normally distributed errors and a GLM of the untransformed data, similarly assuming normally distributed errors but using a log link, were trialled. Note that while both methods use iteratively reweighted least squares (IRLS) to estimate weights, only the log link produces weights that diverge from unity.

As can be seen in Figure A.2 on the facing page, the log link model appears to fit the data reasonably well on a cursory inspection, and it is desirable that the threshold includes many more data points. However, the residuals in Figure A.2(f) reveal strong trends in the data that are divergent from this model (note that residuals are calculated on a log scale in Figure A.2(e); the residuals are actually not too different in magnitude between the two models). This is primarily a result of the log link model heavily weighting against the low magnitude measurements (as evidenced by the IRLS weights; data not shown), which are already of reduced significance since residuals are calculated prior to log transformation of the data. When using the log link with a GLM in the previous analysis in minimal media, it was noted that it produced almost the same result as a non-linear exponential fit of the data, confirming the effect of calculating residuals in the untransformed scaling.

Since the fit is made per well such that successive measurements are correlated, the residual error will not necessarily increase with OD_{600} as it might for a regular assay. For the growth curve assays, the primary source of error is presumably that of the reader or artifacts like bubbles or dust.

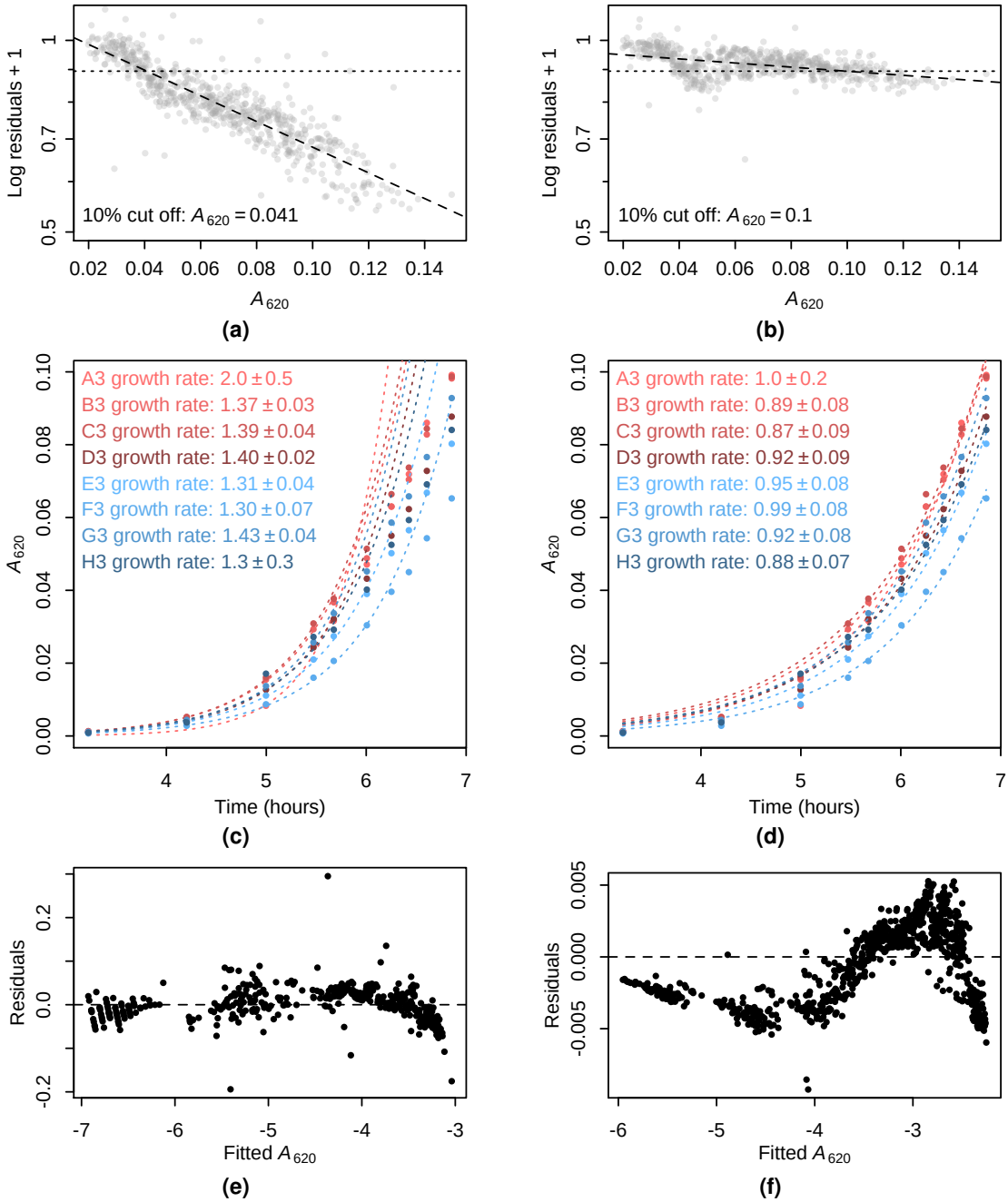


Figure A.2: Fitting the 'lacz2' data using alternative GLMs. The panels on the left are of a GLM with identity link and normally distributed errors, whilst the panels on the right are of a GLM with log link and normally distributed errors. **(a)** & **(b)** Determining an upper threshold based on the fold divergence from ideal exponential growth. Plotted are the residuals for the n^{th} points in the growth series relative to the model fit for the previous $n-1$ points. This was performed for all incremental subsets of the growth data. The divergence was observed to occur exponentially, so the residuals have been translated up by 1 and plotted and fit on a log scale. Note also that for all steps the data set has been truncated for all A_{620}s less than 0.001. **(c)** & **(d)** After specifying a cutoff, the data is fit using this cutoff and example fits are shown. The red curves originate from a high O/N, whilst the blue curves originate from a low O/N. **(e)** & **(f)** Residuals for fits using the upper threshold. Overlaid data from all wells on the plate is shown.

A.2 Gompertz fits

Given the observed divergence from pure exponential growth seen in Appendix A.1, an alternative model might be a more appropriate descriptor. Choosing a model for fitting growth curves is discussed in Zwietering et al. [1990], and they posit the Gompertz model as a balanced, general purpose fit. Their analysis assumes that the data input into each model is the logarithm of the relative population size. They also re-formulate the model in terms of parameters that are relevant to modelling microbial growth — in particular they specify the growth rate as the maximal slope of the log-transformed data. For the Gompertz model, this occurs at the inflection point. Their formulation is:

$$y = A \exp\left[-\exp\left(\frac{\mu e}{A}(\lambda - t) + 1\right)\right], \tag{A.1}$$

where A is the asymptote that the growth curve tends towards, μ is the maximum specific growth rate and λ is the lag time (e is simply $\exp(1)$). We choose to fit this model using the `drc` package, which has both three- and four-parameter implementations:

$$y = C + (D - C) \exp\left[-\exp\left(B(t - E)\right)\right] \tag{A.2a}$$

$$y = D \exp\left[-\exp\left(B(t - E)\right)\right]. \tag{A.2b}$$

By comparing coeffecients inside the exponentials in Equation (A.1) with those in Equations (A.2a) and (A.2b), the growth rate (μ) can be determined from the `drc` fits using $\mu = \frac{-B(D-C)}{e}$, with the lag time given by $\lambda = E + \frac{1}{B}$. The three parameter model is just the case where $C = 0$. Observe that as $t \to \infty$, $y \to D - C$ and as $t \to -\infty$, $y \to C$.

After choosing some initial reference absorbance to calculate the relative population sizes, and then taking the logarithm of the data so that the slope of this transformed growth curve specifies the growth rate at each instant (see Figure A.3), we can observe that the offset coefficient C in the four-parameter Gompertz model would allow correction for errors

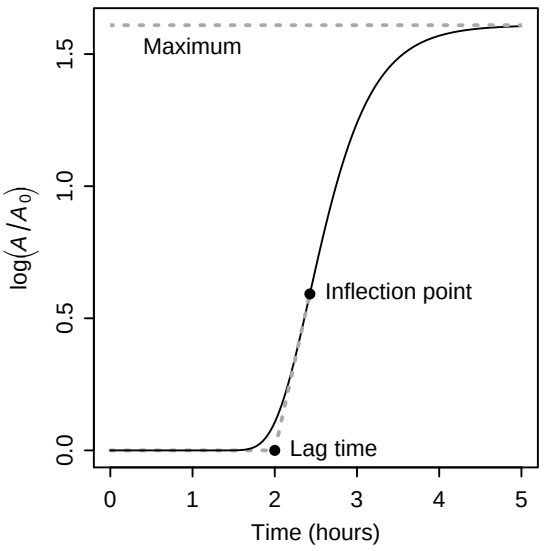


Figure A.3: The Gompertz fit is designed to match data that has a lag period before entering a linear-like region that then asymptotes towards a maximum. As applied to log-transformed growth curves, this matches the masking of growth below instrumental sensitivity during the lag time, an approximately linear slope during the exponential growth phase, followed by a gradual slowing of growth as nutrients are consumed.

in determining the reference absorbance. That is, we have:

$$\log\left(\frac{A(t)}{A_0}\right) = \log\left(A(t)\right) - \log\left(A_0\right) = y - C, \tag{A.3}$$

where $A(t)$ is the time varying absorbance, and A_0 is the absorbance measured at time $t = 0$, so that the parameter $C = \log(A_0)$. Given these observations, it may make the most sense to simply use the initial reading as the per well reference (A_0) instead of using it as an offset.

The Gompertz fitting protocol:

1. Start with a data set using a per-well offset from the zero time point.
2. Choose starting parameters from an intial GLM fit of the data (the GLM fit ignores points with an OD_{600} less than 0.005 or greater than 0.3).
3. Calculate the relative growth using the next positive OD_{600} as the reference.
4. Refine the start parameters using a non-linear least squares Gompertz fit of the data in which the growth rate is set from the GLM fit.
5. Perform a robust Gompertz fit by median estimation.

A.3 Comparing the models

Of the Gompertz models, the three parameter Gompertz model is generally preferred since it has fewer parameters. However, where the reference point is clearly after the lag time point, the extra flexibility of the four parameter model becomes relevant.

Both of the Gompertz models can benefit from an initial estimate of the growth rate by the thresholded log-linear fit. Otherwise, the high-valued points are often over-weighted.

On the whole, for the time courses used for the MFL strains, the three parameter Gompertz model performs the best, producing the smallest standard deviations overall. This is presumably due to the small number of data points in the time course for these experiments, and few readings at early time points, reducing the need for the offset correction introduced by the four parameter Gompertz model. This can be seen in the direct comparison in Figure A.4.

The differences between plates observed across different days were somewhat alarming, especially those observed for the WR-MC and ER-MC strains. The difference in mean in those cases was significantly different (95% confidence limits in the means did not overlap for different days).

The largest differences were observed for the WR-MC strains, but these can apparently be accounted for by the difference in choice of time points. Plate 'wrmc2' has particularly late time points and hence suffers from a poor estimate of the base level reading (see Figure A.5).

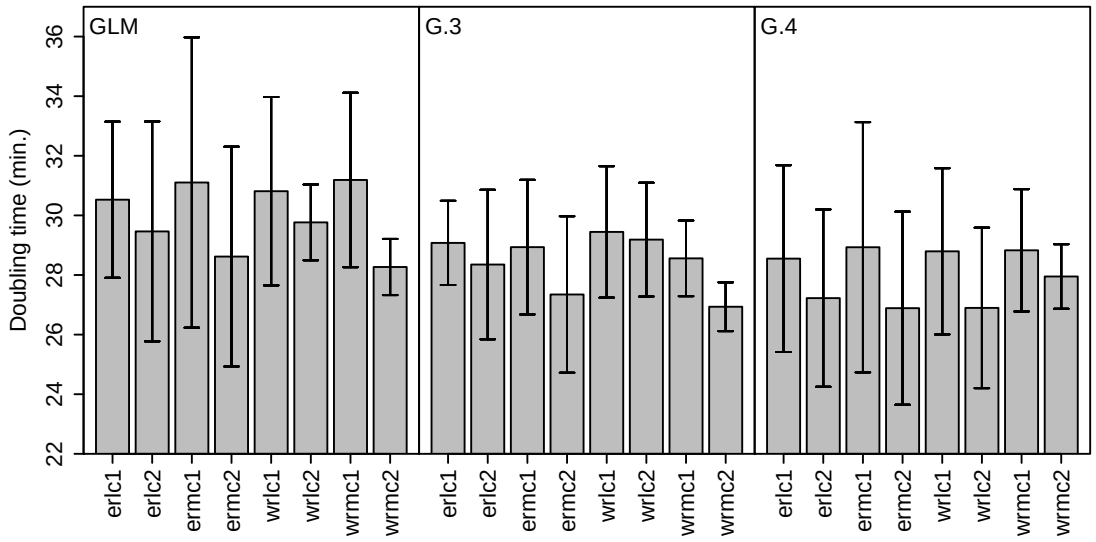


Figure A.4: Comparing determinations of doubling time between the various models of growth for each of the MFL strains. Growth curves measured for the MFL strains were measured over time in 96-well plates during the hysteresis assay, and were fit using either a log-linear model with upper threshold (GLM), a three parameter Gompertz model (G.3), or four parameter Gompartz model (G.4). The doubling times drawn from each of these fits were calculated for each well and the mean value is shown for each plate, with error bars indicating the standard deviation.

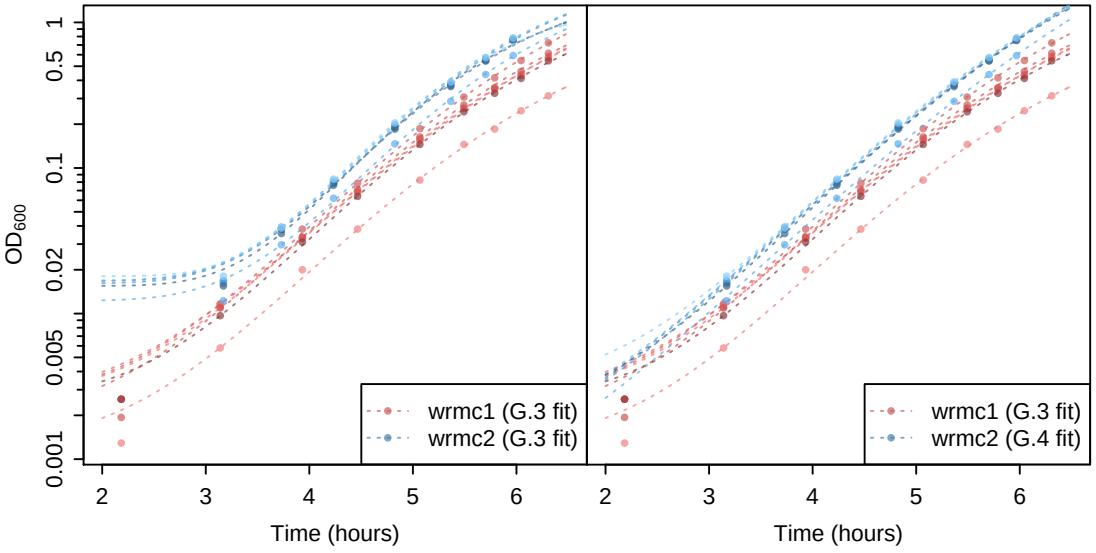


Figure A.5: The 'wrmc2' plate deviates significantly due to fitting biases introduced by the selection of time points. Shown are growth curves and respective Gompertz fits for selected wells of two different hysteresis assay plates, 'wrmc1' and 'wrmc2', showing growth of the WR-MC MFL strain under assay conditions. Three-parameter Gompertz fits (G.3) or four-parameter Gompertz fits (G.4) are shown as indicated.

A.4 Growth rate measurements

The rate of growth of the strains shows little dependence on assay conditions as can be seen for growth rate measurements of the MFL strains shown in Figure A.6 and growth rate measurements of the induction reporter strains shown in Figure A.7.

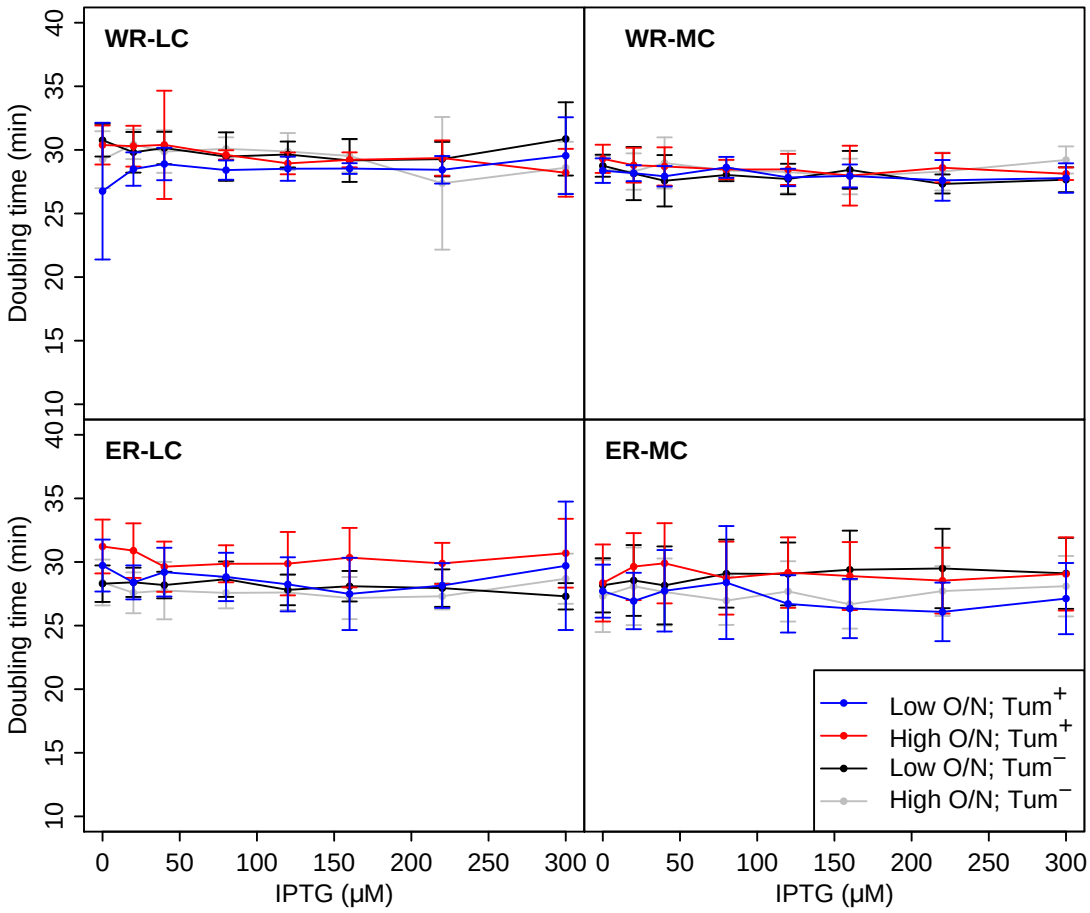


Figure A.6: MFL strain doubling times show little dependence on IPTG or assay conditions. Growth of MFL strains under the hysteresis assay was monitored in 96-well plates by following the A_{620} over time and doubling times derived by fitting those curves with Gompertz models. These are shown plotted versus IPTG and separated by strain and level of induction of the overnight (O/N). Error bars indicate 95% confidence limits on the estimated mean doubling times; $n = 6$ for all data points.

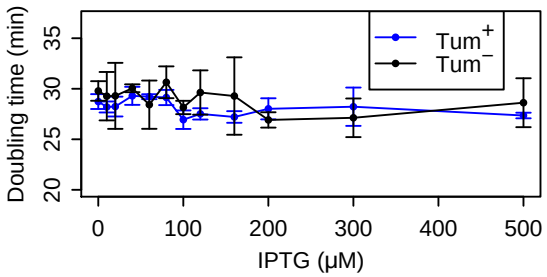


Figure A.7: Induction reporter strain doubling times show little dependence on IPTG or assay conditions. Growth of the induction reporter strains was followed as described in Figure A.6, but the data from the two different O/Ns was aggregated since there were no significant differences between those data sets. Error bars indicate 95% confidence limits on the estimated mean doubling times; $n = 8$ for all data points.

B

Scripts for analysis of flow cytometry data

Flow cytometry data obtained in this thesis was loaded and analysed in R, making use of a number of packages from Bioconductor [Gentleman et al., 2004], including the `flowCore` [Ellis et al., 2009a], and `flowClust` [Lo et al., 2009] packages.

The basic normalisation and curation of each set of data set typically proceeded as listed in the following code excerpt:

```
1  # Load the data sets, transforming only the FSC/SSC channels for cell
2  # population selection:
3  fs.ermc <- fsload('data', logicle=c('FSC.H', 'SSC.A'))
4  fsm.ermc <- SelectCellPop(fs.ermc, saveplot=TRUE)
5
6  # With the main population selected, calculate the Logicle transformation
7  # for the fluorescence channel:
8  fsm.ermc <- MakeLogicle(fsm.ermc, 'FITC.A')
9
10 # Calculate kernel density estimates for the entire flow set:
11 (fsm.ermc.range <- fsRange(fsm.ermc))
12 fsm.ermc.dmaps <- fsDensityMaps(fsm.ermc, fsr=fsm.ermc.range)
13
14 # Apply the morphology regression and normalisation:
15 fsn.ermc <- fsApply(fsm.ermc, morphNormalise, densitymaps=fsm.ermc.dmaps,
16                     fluorchannel='FITC.A.LOG')
17
18 # Annotate and order the flow set:
19 maskdict <- MaskDictionary()
20 fsna.ermc <- fsParseNames(fsn.ermc, translate2009, ordering2009, maskdict)
21 pData(fsna.ermc) # check the annotation
22
23 # Calculate unimodal Skew.t fits of all the control curves as a training set
24 fits.ermc.ctrl <- skewt.unifit(fsna.ermc[pData(fsna.ermc)$ctrl=='TM'])
25
26 # Store a list of models that can be used to predict skew t parameters from
27 # a median:
28 skewt.predictor.ermc <- skewt.predictor(fits.ermc.ctrl)
```

```
29
30 modalfits.ermc <- ModalityAnalysis(fsna.ermc, skewt.predictor.ermc)
31
32 fitsummary.ermc <-
33   cbind(pData(fsna.ermc), ciprod=normER(pData(fsna.ermc)),
34         fsStats(fsna.ermc, channels='FITC.A.LOG.M'),
35         summary(modalfits.ermc, W=getWparam(fsna.ermc),
36                 predictor=skewt.predictor.ermc))
```

The most important methods from that excerpt will be described in more detail in the remainder of this appendix. These include automated selection of the main cell population (Appendix B.2), Logicle transformation of the data (Appendix B.3), morphology normalisation of the data (Appendix B.4), and constrained skew-*t* regression of the MFL samples (Appendix B.5).

B.1 General utility functions

Some general utility functions were used in many of the following protocols. These functions are listed here for reference.

```
1  ##########################################################################
2  ### Utility functions and distributions for the fcanalysis library ###
3  ##########################################################################
4
5  # Skew-t distribution
6  dskewt <- function(x,params,n=1){
7    if(!require(sn, quietly=TRUE))
8      stop("the sn package must be installed to calculate skew-t functions")
9    params$pii[n]*dst(x,
10     xi=params$mu[n],
11     omega=sqrt(params$sigma2[n]),
12     alpha=params$shape[n],
13     nu=params$nu[n])
14 }
15
16 # Bimodal skew-t distribution
17 dskewt.mix <- function(x,params){
18   nmix <- length(params$mu)
19   y <- numeric(length(x))
20   for(i in 1:nmix)
21     y <- y+dskewt(x,params=params,n=i)
22   return(y)
23 }
24
25 # Logicle functions {{{
```

```
26
27 calcWfromP <- function(P) 2*P*log10(P)/(P+1)
28
29 Logicle <- function(x, W, M=4.5, T=262144, P=NA){
30   if(is.na(P))
31     P <- uniroot(function(p){W-calcWfromP(p)}, c(0.1,100))$root
32   lt.W <- x<W
33   x[lt.W]<- -T*10^(-(M-W))*(10^(-x[lt.W]+W)-P^2*10^(-(-x[lt.W]+W)/P)+P^2-1)
34   x[!lt.W]<- T*10^(-(M-W))*(10^(x[!lt.W]-W)-P^2*10^(-(x[!lt.W]-W)/P)+P^2-1)
35   x
36 }
37
38 invLogicle <- function(S, W, M=4.5, T=262144, P=NA){
39   if(is.na(P))
40     P <- uniroot(function(p){W-calcWfromP(p)}, c(0.1,100))$root
41
42   loglim <- Logicle(2*W, W=W, M=M, T=T, P=P)
43
44   sapply(S, function(s){
45           if(s>loglim) lims <- c(2*W, log10(s*10^(M-W)/T)+W)
46           else if(s>0) lims <- c(W,2.1*W)
47           else if(s>-loglim) lims <- c(0, 1.1*W)
48           else lims <- c(-log10(-s*10^(M-W)/T)+W, 0.1)
49
50           uniroot(function(x){s-Logicle(x, W=W, M=M, T=T, P=P)},
51                   lims)$root
52     })
53 }
54
55 LogicleWidth <- function(NegRef, M=4.5, T=262144)
56   (M-log10(T/abs(NegRef)))/2
57
58 LogicleNegRef <- function(W, M=4.5, T=262144)
59   -T*10^(2*W-M)
60
61 #}}}
62
63 # Curve Fitting Functions {{{
64
65 RChiS <- function(Func,p,datax,datay){
66   return(sum(
67       (datay-Func(datax,p))^2/Func(datax,p)
68     )/(length(datax)-length(p)))
69 }
```

```
70
71 LeastSquare <- function(p,Func,datax,datay){
72   return(sum((datay-Func(datax,p))^2))
73 }
74
75 #}}}
```

B.2 Automated selection of the main cell population

The main cell population was selected with the aid of the `flowClust` package and the following functions:

```
1  SelectCells <- function(ff, level=0.8, channels=c('FSC.H.LOG', 'SSC.A.LOG')){
2    cluster <- flowClust(ff, varNames=channels, K=1, B=100)
3    ruleOutliers(cluster) <- list(level=level)
4    return(list(ffC=ff[ff %in% cluster,], cluster=cluster))
5  }
6
7  # Select the main cell population using a generous threshold, since the
8  # FSC/SSC data can later be used to normalise the fluorescence data according
9  # to morphology. SelectCellPop can also optionally save a plot of the chosen
10 # region.
11 SelectCellPop <- function(x, ...) UseMethod('SelectCellPop')
12 SelectCellPop.default <- function(x, ...)
13   stop('"SelectCellPop" not implemented for this class.')
14
15 SelectCellPop.flowFrame <- function(x, level=0.95, ...){
16     ffcluster <- SelectCells(x, level=level)
17     return(ffcluster$ffC)
18   }
19
20 SelectCellPop.flowSet <- function(x, channels=c('FSC.H.LOG','SSC.A.LOG'),
21                                   saveplot=FALSE, xlim=NULL, ylim=NULL, ...){
22   if(saveplot){
23     ranges <- apply(fsApply(x[,channels], each_col, quantile,
24                             probs=c(0.005,0.99)), 2, range)
25
26     if(is.null(xlim)){
27       Wparam.x <- getWparam(x, channels[[1]])
28       if(!is.null(Wparam.x))
29         xlim <- Logicle(ranges[,channels[[1]]], Wparam.x)
30       else xlim <- ranges[,channels[[1]]]
31     }
32
```

```
33      if(is.null(ylim)){
34        Wparam.y <- getWparam(x, channels[[2]])
35        if(!is.null(Wparam.y))
36          ylim <- Logicle(ranges[,channels[[2]]], Wparam.y)
37        else ylim <- ranges[,channels[[2]]]
38      }
39    }
40
41    fsApply(x, SelectCellPop.flowFrame, channels=channels, ...)
42  }
```

B.3 Logicle transformation

For Logicle transformation, a common W parameter was chosen according to the method specified by Parks et al. [2006], and this parameter was stored as an attribute for each `flowFrame` (`ff`). The transformation with this parameter was then applied to a specified flow cytometry channel using the `logicleTransform` function of the `flowCore` package. The following functions were used to automate this process.

```
1  setWparam <- function(ff, channel, W){
2    if(!all(grepl('\\.LOG[.]?.*$', channel)))
3      stop('Channel name has incorrect format: it should include ".LOG"')
4    eval(parse(text=paste0('keyword(ff)<-list(', channel, '.W=W)')))
5    return(ff)
6  }
7
8  getWparam <- function(ff, channel='FITC.A.LOG'){
9    keyname <- sub('\\.LOG[.]?.*$', '.LOG.W', channel)
10   if(inherits(ff, 'flowFrame'))
11     return(keyword(ff, keyname)[[1]])
12   if(inherits(ff, 'flowSet'))
13     return(unique(fsApply(ff, function(ffi) keyword(ffi,keyname)[[1]])[,1]))
14   return(NULL)
15 }
16
17 MakeLogicle <- function(fs, channel="FSC.A"){
18   # To calculate the width parameter, choose the fifth percentile of all
19   # negative values in the data set as suggested in parks2006nld
20   fsNegRef <-
21     quantile(as.vector(fsApply(fs, function(ff)
22                                  exprs(ff[,channel][ff[,channel]<0]) )),
23              probs=0.05)[[1]]
24   if(is.na(fsNegRef) || fsNegRef>-10) fsNegRef <- -10
25   cat(paste('Choosing', fsNegRef, 'as negative reference for', channel, '\n'))
```

```
26
27    srcname <- channel
28    newname <- paste(srcname, "LOG", sep=".")
29
30    fsW <- (4.5-log10(262144/abs(fsNegRef)))/2
31    # Parameters for the transformation function are mostly the defaults, but
32    # set specifically just in case the defaults change:
33    assign('fcanalysis.logicleFunc',
34           logicleTransform(transformationId='fcanalysisLogicleTransform',
35                            w=fsW, t=262144, m=4.5, a=0),
36           globalenv()) # transform function needs to be in global name space
37
38    # Set up the transformation with correct channel names and call it
39    fs <-
40      eval(parse(text=paste0('transform(fs,`',newname,
41                             '`=fcanalysis.logicleFunc(`',srcname,'`))')))
42
43    fsApply(fs, setWparam, channel=newname, W=fsW)
44  }
```

B.4 Morphology normalisation

Two alternative protocols for morphology normalisation were used in this thesis: normalisation using an average cell density (as per the protocol specified by [Knijnenburg et al., 2011]), or normalisation using a sample-specific cell density. Morphology normalisation using an average cell density works well for the majority of the Tum−CI MFL data, but substantial biases in absolute intensities were introduced for many of the high intensity samples. An example is the deviation of the high fluorescence data points (particularly in the controls) from the respective LacZ assay points back in Figure 5.3(a). Though it may seem minor, the accuracy of the normalisation protocol for a majority of the assays became suspect since the raw fluorescence means better matched the LacZ assay results. The errors were found to be associated with the conversion from relative to absolute intensities in the last step of the normalisation protocol, where an 'average' cell density in FSC/SSC space is used to estimate the true fluorescence mean from the fitted regression surface. Whilst the 'average' cell density worked well for a few high quality data sets, for the majority, the low representation of high fluorescence samples in the average density map amplified errors in poorly constrained regions of their regression surfaces. Since the absolute mean fluorescence of the raw data was a more reliable indicator of true mean activity, for most assays, the sample-specific FSC/SSC density (derived from the two-dimensional kernel density estimate of the FSC/SSC for that particular sample) was instead used to calculate the expected fluorescence for each regression surface. The effect of morphology normalisation

by the sample-specific density can be seen back in Figure 5.3(b). Whilst discrepancies with the LacZ assay remain, they are smaller in comparison with normalisation by average cell density and better reflect the expected trends.

Knijnenburg et al. developed the method for deriving the representative cell density by analogy with standard flow cytometry analyses, where fair inter-sample fluorescence comparisons are obtained by maximising population homogeneity using a stringent but consistent filter (gate) over the FSC and SSC channels. Knijnenburg et al. [2011] calculate a FSC/SSC density map that represents the 'average' cell morphology for an entire comparison group, and use this to weight the calculation of expected (mean) fluorescence for each regression surface.

The largest deviations tended to arise for samples whose FSC/SSC density map was poorly represented in the average density map. Poor representation resulted in incorrect calculation of the expected fluorescence for such regression surfaces, since low-confidence regions of the extrapolated surface were more heavily weighted, amplifying any inaccuracies in those regions. In other words, the average cell morphology had high densities in regions of the FSC/SSC space where the most problematic samples had few data points and thus where the regression surface was poorly constrained. That is, the errors were associated with poor inter-sample FSC/SSC overlap that did not appear to be correlated with fluorescence.

For the three high quality assays, the original normalisation using average cell density did not introduce errors and, in fact, improved the results. This was attributed both to higher cell counts (around 80,000 — at least twice as many as previously measured) and to greater care taken to ensure stability of FSC and SSC distributions during collection of the measurements. These experimental changes were prompted by observations of FSC/SSC drift occurring over the course of an experiment, presumably caused by occasional instabilities in the flow rate. Monitoring for such drift and increasing the overlap of FSC/SSC density using higher cell counts is recommended for future assays where the intent is to apply morphology normalisation. In this thesis, these three high quality assays are left normalised using the average cell density. For all other assays, normalisation of the fluorescence intensity was performed using the morphology normalisation protocol with sample-specific densities.

The protocol used by [Knijnenburg et al., 2011] was written in Matlab. For convenience, the protocol was rewritten in R for this thesis as listed in the following code block.

```
1  # Calculate cell density maps for each flow frame
2  fsDensityMaps <- function(fs, channels=c("FSC.H","SSC.A"),
3                            fsr=fsRange(fs,channels), n=2^8){
4
5    if(!require(MASS, quietly=TRUE))
6      stop("the MASS package must be installed to calculate kernel density estimates")
7
```

```
8   # Obtain the necessary flow set limits
9   lims <- list()
10  for(i in channels)
11    lims[[i]] <- range(fsr[,paste(i, c("MIN","MAX"), sep=".")])
12
13  # Calculate a density map for each flow frame
14  densitymaps <-
15    fsApply(fs,function(ff){
16            cat('.') # track progress - this calculation can take a while
17            return(kde2d(exprs(ff[,channels[1]]), exprs(ff[,channels[2]]),
18                         n=n, lims=unlist(lims)))
19                          })
20  # NB: The kde2d function comes from the MASS package.
21  cat('\n')
22
23  # Normalise the densities
24  lapply(densitymaps, function(dens){
25         dens$z <- dens$z/sum(dens$z)
26         return(dens)})
27  }
28
29  # Calculate the average cell density for the entire flow set
30  fsAverageDensity <- function(densitymaps){
31    # Sum all the densities together
32    totaldensity <-
33      Reduce(function(total,addmap) within(total, z <- z + addmap$z),
34             densitymaps[-1], densitymaps[[1]])
35    totaldensity$z <- totaldensity$z/length(densitymaps)
36    return(totaldensity)
37  }
38
39  # Make a grid of constraints to force the regression model to remain
40  # monotonic. This function is called by 'ffRegressConstrained' and requires a
41  # model that is of class 'gam'.
42  makeConstraintGrid <- function(model, lims, channels=c("FSC.H","SSC.A"), n=2^3){
43    # Create two offset coordinate grids with the specified dimensions:
44    fscrow <- seq(lims[[channels[1]]][1], lims[[channels[1]]][2], length.out=n)
45    ssccol <- seq(lims[[channels[2]]][1], lims[[channels[2]]][2], length.out=n)
46    lowgrid<-data.frame(
47      fsc=c(rep(fscrow[-n],times=n),rep(fscrow,times=n-1)),
48      ssc=c(rep(ssccol,each=n-1),rep(ssccol[-n],each=n))
49    )
50    highgrid<-data.frame(
51      fsc=c(rep(fscrow[-1],times=n),rep(fscrow,times=n-1)),
```

```
52      ssc=c(rep(ssccol,each=n-1),rep(ssccol[-1],each=n))
53    )
54
55    # At these grid locations, determine a linear predictor for the specified
56    # model. The linear predictor is a matrix that can be used to calculate the
57    # value of the regression model at these locations by post-multiplying with
58    # the parameter vector.
59    X0 <- predict(model, newdata=lowgrid, type="lpmatrix")
60    X1 <- predict(model, newdata=highgrid, type="lpmatrix")
61    constrgrid <- X1-X0
62    # Attach the grid positions that were used to facilitate plotting:
63    attr(constrgrid, "fscrow") <- fscrow
64    attr(constrgrid, "ssccol") <- ssccol
65    return(constrgrid)
66  }
67
68  # This function fits fluorescence data to cell morphology data (FSC/SSC) using
69  # the constrained regression model suggested by knijnenburg2011rma. The
70  # monotonic constraint is applied using a coarse grid (constructed by
71  # 'makeConstraintGrid') and constrained least squares fitting is performed by
72  # the 'pcls' function of the 'mgcv' package. The function is based on the
73  # example of monotonic regression given in the 'pcls' documentation.
74  ffRegressConstrained <- function(ff,lims,scatterchannels=c("FSC.H","SSC.A"),
75      fluorchannel="FITC.A"){
76    if(!require(mgcv, quietly=TRUE))
77      stop('the "mgcv" package must be installed to perform constrained regression')
78
79    ff.df<-data.frame(
80      fsc=exprs(ff[,scatterchannels[1]])[,1],
81      ssc=exprs(ff[,scatterchannels[2]])[,1],
82      fl=exprs(ff[,fluorchannel])[,1]
83    )
84
85    # Use 'gam' from the 'mgcv' package to set up the design matrix for the GLM
86    # that can be input into the 'pcls' function after adding the constraints.
87    G <- gam(fl~(fsc+ssc)^2+(sqrt(fsc)+sqrt(ssc))^2+I(fsc^2)+I(ssc^2),
88             data=ff.df, fit=FALSE)
89    # Perform a preliminary unconstrained fit:
90    prelimfit <- gam(G=G)
91
92    # Enforce monotonicity by first calculating the finite difference between
93    # two grids and constraining these to all be greater than zero by defining
94    # the inequality matrix (Ain).
95    G$Ain <- makeConstraintGrid(prelimfit, lims=lims, channels=scatterchannels)
```

```
96  G$bin <- rep(0,nrow(G$Ain))
97
98  # Force initial parameters to match the constraint:
99  G$p <- coef(prelimfit)
100 G$p[G$p<0] <- 0
101
102 # Fit the coefficients using the 'penalised constrained least squares
103 # fitting' function 'pcls':
104 finalfit <- prelimfit
105 finalfit$coefficients <- pcls(G)
106 if(any(is.na(finalfit$coefficients))){
107   warning('some coefficients for ', keyword(ff, 'TUBE NAME')[[1]],
108           ' are NA; reattempting with alternative initial parameters')
109
110   # Start the initial fit again with a smaller subset of the parameters
111   # initG <- gam(fl~fsc+ssc+sqrt(fsc)+sqrt(ssc)+I(fsc^2)+I(ssc^2),
112   #              data=ff.df, fit=FALSE)
113   initG <- gam(fl~fsc+ssc+sqrt(fsc)+sqrt(ssc),
114                data=ff.df, fit=FALSE)
115   initfit <- gam(G=initG)
116
117   G$p <- rep(0, length(G$term.names))
118   names(G$p) <- G$term.names
119   G$p[names(coef(initfit))] <- coef(initfit)
120   G$p[G$p<0] <- 0
121
122   finalfit$coefficients <- pcls(G)
123   if(any(is.na(finalfit$coefficients)))
124     warning('regression for ', keyword(ff, 'TUBE NAME')[[1]], ' failed')
125 }
126
127 # Check that all constraints are satisfied:
128 constraints <- G$Ain %*% finalfit$coefficients
129 if(any(constraints <= 0))
130   warning('Regression for ', keyword(ff, 'TUBE NAME'),
131           ' breaks ', sum(constraints <= 0), ' of ',
132           length(constraints), ' constraints with a total deviation of ',
133           format(sum(constraints[constraints <= 0])))
134
135 # Update useful additional properties of the fit:
136 names(finalfit$coefficients) <- names(G$p)
137 finalfit$fitted.values <- predict(finalfit)
138 finalfit$residuals <- finalfit$y-finalfit$fitted.values
139
```

```
140   return(finalfit)
141 }
142
143 # Normalise the fluorescence channel of a flow frame according to its
144 # 'morphology' (i.e., FSC and SSC). The algorithm prescribed in
145 # knijnenburg2011rma is followed.
146 morphNormalise <- function(ff, densitymaps, scatterchannels=c("FSC.H","SSC.A"),
147                            fluorchannel="FITC.A", dmap.index=NULL){
148   if(!require(mgcv, quietly=TRUE))
149     stop('the "mgcv" package must be installed to perform constrained regression')
150
151   if(is.null(dmap.index)){
152     # Either the function is already supplied with the correct cell density:
153     if(length(densitymaps)==3L && all(names(densitymaps)[1:3]==c('x','y','z')))
154       celldensity <- densitymaps
155     # Or else it is supplied with a list of density maps to calculate the
156     # average from:
157     else
158       celldensity <- fsAverageDensity(densitymaps)
159   } else {
160     # Use the 'dmap.index' to specify a dmap:
161     celldensity <- densitymaps[[keyword(ff, dmap.index)[[1]]]]
162   }
163
164   lims <- list()
165   lims[[scatterchannels[1]]] <- range(celldensity$x)
166   lims[[scatterchannels[2]]] <- range(celldensity$y)
167
168   constrfit <-
169     ffRegressConstrained(ff, lims=lims, scatterchannels=scatterchannels,
170                          fluorchannel=fluorchannel)
171
172   densitygrid <- data.frame(
173     fsc=rep(celldensity$x, times=length(celldensity$y)),
174     ssc=rep(celldensity$y, each=length(celldensity$x))
175   )
176   densitygrid$fl <- predict(constrfit, newdata=densitygrid)
177   celldensity$z.flat <- as.numeric(celldensity$z)
178   avgfl <- sum(celldensity$z.flat*densitygrid$fl)
179
180   normfl <- avgfl+residuals(constrfit)
181   normfl <- matrix(normfl, nrow=length(normfl), ncol=1,
182                    dimnames=list(NULL, paste(fluorchannel,"M",sep=".")) )
183
```

```
184   return(cbind2(ff, normfl))
185 }
```

B.5 Constrained skew-*t* regression

Unimodal skew-*t* distributions were initially fit to the control samples in an unconstrained manner using the `mixsmsn` package [Prates et al., 2013]. This was applied to a `flowSet` using the following function

```
1  # Generate unimodal Skew.t fits over a flowSet as training data {{{
2  # NB: in order to save some time, the following function does not calculate
3  # the information matrix.
4  skewt.unifit <- function(fs, channel='FITC.A.LOG.M'){
5    if(!require(mixsmsn, quietly=TRUE))
6      stop("the mixsmsn package must be installed to fit skew-t mixture models")
7
8    fits <-
9      fsApply(fs, function(ff){
10             cat('.') # Progress counter since this can be slow
11             # Calculate some statistics
12             ffdata <- as.vector(exprs(ff[,channel]))
13             ffquant <- quantile(ffdata, probs=c(0.25,0.5,0.75), na.rm=TRUE)
14             ffmean <- mean(ffdata, na.rm=TRUE)
15             ffvar <- sd(ffdata, na.rm=TRUE)^2
16
17             unifit <-
18               smsn.mix(ffdata, g=1, family='Skew.t', pii=1,
19                        nu=5, mu=ffquant[[2]], sigma2=ffvar, shape=-1,
20                        get.init=FALSE, calc.im=FALSE)
21
22             list(unifit=unifit,
23                  stats=list(quant=ffquant,mean=ffmean,var=ffvar))
24     })
25   cat('\n')
26   structure(fits, class=c('skewt.unifit','list'))
27 }
28
29 print.skewt.unifit <- function(x,...)
30   cat(paste0('A list of unimodal Skew.t fits of length ',
31              length(x),'\n'))
32
33 summary.skewt.unifit <- function(x,...){
34   fitsummary <-
35     sapply(x, function(fit)
```

```
36              c(mu=fit$uni$mu, sigma2=fit$uni$sigma2, nu=fit$uni$nu,
37                shape=fit$uni$shape, quant25=fit$stats$quant[[1]],
38                quant50=fit$stats$quant[[2]], quant75=fit$stats$quant[[3]],
39                mean=fit$stats$mean, var=fit$stats$var))
40    fitsummary <- as.data.frame(t(fitsummary))
41    within(fitsummary, quantdiff <- quant75-quant25)
42  }
43
44  #}}}
```

A 'predictor' object was constructed from these fitted skew-*t* distributions by calculating a series of regression models that described the data. These regression models are saved and enable the prediction of all other skew-*t* parameters from just a supplied median fluorescence. The following code shows how this was implemented.

```
1   # Generate a 'skewt.predictor' object from unimodal Skew.t fits by
2   # constructing a catalog of regression models for each Skew.t parameter
3   # against the median value determined from the training set. This object can
4   # then later be used to predict skew t parameters from a median:
5   skewt.predictor <- function(unifit){
6     if(!inherits(unifit, 'skewt.unifit'))
7       stop('"unifit" must be a "skewt.unifit" object')
8
9     unifitsummary <- summary(unifit)
10
11    predictor <-
12      list(mu.median=lm(mu~quant50, data=unifitsummary),
13           width.median=lm(quantdiff~quant50+I(quant50^2), data=unifitsummary),
14           sigma2.width=lm(sigma2~quantdiff+I(quantdiff^2), data=unifitsummary),
15           shape.width=lm(shape~quantdiff+I(quantdiff^2), data=unifitsummary),
16           nu.sigma2=lm(nu~sigma2+I(sigma2^2), data=unifitsummary))
17    structure(predictor, class=c('skewt.predictor', 'list'))
18  }
19
20  # The following function estimates skew t parameters for a specified median
21  # fluorescence, and optionally additionally a peak width, which is here
22  # defined as the difference between the 25th and 75th percentiles. It returns
23  # a 'Skew.t' object describing a mixture of skew t models.
24  predict.skewt.predictor <- function(object, median, width=NULL, pii=1,
25                                      checks=TRUE){
26    if(missing(median))
27      stop('At least one "median" value must be supplied for prediction')
28
29    if(is.null(width))
30      width <- predict(object$width.median, newdata=data.frame(quant50=median))
```

```
31
32    if(checks){
33      if(length(median) != length(width) || length(median) != length(pii))
34        stop('The specified medians, widths and probabilities must all have ',
35             'the same lengths')
36
37      if(sum(pii) != 1.0)
38        stop('Mixture component probabilities "pii" must sum to 1')
39    }
40
41    sigma2 <- predict(object$sigma2.width, newdata=data.frame(quantdiff=width))
42
43    pred <-
44      list(pii=pii,
45           mu=predict(object$mu.median, newdata=data.frame(quant50=median)),
46           sigma2=sigma2,
47           shape=predict(object$shape.width, newdata=data.frame(quantdiff=width)),
48           nu=predict(object$nu.sigma2, newdata=data.frame(sigma2=sigma2)))
49
50    # For predicted values of 'nu' less than 1, take an average value instead to avoid
51    # small/negative degrees of freedom
52    if(any(pred$nu < 1))
53      pred$nu[pred$nu<1] <- mean(object$nu.sigma2$model$nu)
54
55    structure(as.data.frame(pred), class=c('Skew.t','data.frame'))
56  }
```

Maximum likelihood estimation could then be used to optimise the median fluorescence to fit the MFL distributions with either a unimodal or bimodal mixture model. This was implemented using the following code.

```
1  # Perform maximum-likelihood estimation to fit a unimodal skew t distribution
2  # that is restricted to the family defined by a skewt.predictor object.
3  unimodal.skewt.em <- function(ff, predictor, channel='FITC.A.LOG.M'){
4    if(!inherits(ff, 'flowFrame')) stop('"ff" must be a "flowFrame"')
5    if(!inherits(predictor, 'skewt.predictor'))
6      stop('"predictor" must be a "skewt.predictor"')
7
8    ffdata <- exprs(ff[,channel])
9    medrange <- c(0.7,1.3)*range(predictor$mu.median$model$quant50)
10
11   fit <-
12     optim(c(med=median(ffdata)),
13           function(pars){
14             probs <- with(as.list(pars),
```

```
15                          dskewt(ffdata, predict(predictor, median=med)))
16            # Avoid an NaN likelihood by setting all ill-behaved probabilities
17            # to be suitably improbable:
18            if(any(is.na(probs) | probs<=0))
19              probs[is.na(probs) | probs<=0] <- 1e-8
20            # Return the log likelihood; NB: optim minimises by default
21            return(-sum(log(probs)))
22          }, method='L-BFGS-B', hessian=TRUE,
23          lower=medrange[[1]], upper=medrange[[2]])
24
25    structure(c(list(skewt.pars=predict(predictor, median=fit$par[['med']])),
26                fit), class='unimodal.fit')
27  }
28
29  # Perform maximum-likelihood estimation to fit a bimodal mixture of skew t
30  # distributions that have been restricted to the family defined by a
31  # skewt.predictor object.
32  bimodal.skewt.em <- function(ff, predictor, channel='FITC.A.LOG.M'){
33    if(!inherits(ff, 'flowFrame')) stop('"ff" must be a "flowFrame"')
34    if(!inherits(predictor, 'skewt.predictor'))
35      stop('"predictor" must be a "skewt.predictor"')
36
37    ffdata <- exprs(ff[,channel])
38    ffquant <- quantile(ffdata, probs=c(0.25, 0.75))
39    medrange <- c(0.7,1.3)*range(predictor$mu.median$model$quant50)
40
41    # Optimise the expectation over the following variables:
42    # - p.lo (the proportion of the low fluorescence population)
43    # - m.lo (the median of the low fluorescence population)
44    # - m.hi.fac (a multiplication factor giving the median of the high
45    #   fluorescence population relative to 'med.low')
46
47    fit <-
48      optim(c(p.lo=0.5, m.lo=ffquant[[1]], m.hi.fac=ffquant[[2]]/ffquant[[1]]),
49            function(pars){
50              # Don't check parameters in predict function during optimisation
51              predpars <- with(as.list(pars),
52                               predict(predictor, median=c(m.lo, m.hi.fac*m.lo),
53                                       pii=c(p.lo, 1-p.lo), checks=FALSE))
54              probs <- dskewt.mix(ffdata, predpars)
55              # Avoid an NaN likelihood by setting all ill-behaved probabilities
56              # to be suitably improbable:
57              if(any(is.na(probs) | probs<=0))
58                probs[is.na(probs) | probs<=0] <- 1e-8
```

```
59              # Return the log likelihood; NB: optim minimises by default
60              return(-sum(log(probs)))
61            }, method='L-BFGS-B', hessian=TRUE,
62            lower=c(0, medrange[[1]], 1.05),
63            upper=c(1, medrange[[2]], medrange[[2]]/medrange[[1]]))
64
65    predpars <- with(as.list(fit$par),
66                     predict(predictor, median=c(m.lo, m.hi.fac*m.lo),
67                             pii=c(p.lo, 1-p.lo)))
68    structure(c(list(skewt.pars=predpars), fit), class='bimodal.fit')
69  }
```

Finally, the constrained mixture modelling analysis was performed over a `flowSet` using the `ModalityAnalysis` function, and the integrated mean values and parameter errors calculated using the `summary` function for a `modality.fits` object. The errors in the fit parameters were calculated from the Hessian output of the `optim` function and were transformed into the appropriate derived parameters (i.e., the distribution means) using the Jacobian.

```
1   ModalityAnalysis <- function(fs, predictor, channel='FITC.A.LOG.M'){
2     if(save){
3       xlim <- range(fsApply(fs, function(ff) range(exprs(ff)[,channel])))
4       W <- getWparam(fs, channel)
5       if(!is.null(W))
6         xlim <- Logicle(xlim, W)
7     }
8
9     fits <-
10      fsApply(fs, function(ff){
11              cat('.')
12              unimodal <-
13                unimodal.skewt.em(ff, predictor=predictor, channel=channel)
14              bimodal  <-
15                bimodal.skewt.em(ff, predictor=predictor, channel=channel)
16              list(unimodal=unimodal, bimodal=bimodal)
17                    })
18    cat('\n')
19    structure(fits, class='modality.fits', channel=channel)
20  }
21
22  summary.modality.fits <- function(x, W=NULL, predictor=NULL, ...){
23    summarise.modality <- function(fits){
24      uniskt <- fits$unimodal$skewt.pars
25      biskt <- fits$bimodal$skewt.pars
26
```

```
27  # Find the mean value of each skew t probability distribution:
28  intskt <- function(pars, n)
29    integrate(function(x) x*dskewt(x, pars, n=n), -Inf, Inf)$value
30
31  # For calculating means, set probability of each distribution to unity:
32  biskt$pii <- rep(1,nrow(biskt))
33
34  uniskt.mean <- c(mean.uni=intskt(uniskt, n=1))
35  lowskt.mean <- c(mean.low=intskt(biskt, n=1))
36  highskt.mean <- c(mean.high=intskt(biskt, n=2))
37
38  if(!is.null(W)){
39    # If W parameter is given, also calculate the distribution means in the
40    # unlogged scale. This requires a change of variables, and hence the
41    # Jacobian (gradient) of the invLogicle function. The 'grad' function
42    # comes from package 'numDeriv':
43    if(!require(numDeriv, quietly=TRUE))
44      stop("the numDeriv package must be installed to calculate ",
45           "skew t distribution means in the untransformed scale")
46
47    # Can't perform Logicle transformation at infinity, so choose suitably
48    # large bounds instead:
49    intBounds <- c(-2.5,4.5)
50
51    cat('.') # This can be slow so track progress
52
53    P <- uniroot(function(p) W - calcWfromP(p), c(0.1, 100))$root
54
55    intskt.unlog <- function(pars, n)
56      tryCatch(integrate(function(x) Logicle(x,W=W,P=P)*dskewt(x,pars,n=n),
57                         intBounds[[1]], intBounds[[2]])$value,
58               error=function(e){ warning(e); return(NA) })
59
60    uniskt.mean <- c(uniskt.mean, rawmean.uni=intskt.unlog(uniskt,n=1))
61    lowskt.mean <- c(lowskt.mean, rawmean.low=intskt.unlog(biskt,n=1))
62    highskt.mean <- c(highskt.mean, rawmean.high=intskt.unlog(biskt,n=2))
63
64    # If the predictor is supplied, calculate standard deviations on the
65    # estimated raw means:
66    if(!is.null(predictor)){
67      # Define functions to allow parameter variation:
68      calcRawMeanUni <- function(pars){
69        skewtpars <- predict(predictor, median=pars[['med']])
70        intskt.unlog(skewtpars, n=1)
```

```
71          }
72          calcRawMeanLow <- function(pars){
73            skewtpars <- predict(predictor, median=pars[['m.lo']])
74            intskt.unlog(skewtpars, n=1)
75          }
76          calcRawMeanHigh <- function(pars){
77            skewtpars <-
78              predict(predictor, median=pars[['m.lo']]*pars[['m.hi.fac']])
79            intskt.unlog(skewtpars, n=1)
80          }
81
82          uni.par <- fits$unimodal$par
83          uni.vcov <- solve(fits$unimodal$hessian)
84          bi.par <- fits$bimodal$par
85          bi.vcov <- solve(fits$bimodal$hessian)
86          bi.piisd <- drop(sqrt(bi.vcov['p.lo','p.lo']))
87
88          # Accuracy is not as important for the errors so use the 'simple'
89          # method for the gradient:
90          uni.jac <-
91            grad(calcRawMeanUni, uni.par['med'], method='simple')
92          bi.lowjac <-
93            grad(calcRawMeanLow, bi.par['m.lo'], method='simple')
94          bi.highjac <-
95            grad(calcRawMeanHigh, bi.par[c('m.lo', 'm.hi.fac')], method='simple')
96
97          uni.sd <- drop(sqrt(uni.jac %*% uni.vcov['med','med'] %*% uni.jac))
98          bi.lowsd <- sqrt(bi.lowjac %*% bi.vcov['m.lo','m.lo'] %*% bi.lowjac)
99          bi.highsd <- sqrt(bi.highjac %*% bi.vcov[c('m.lo', 'm.hi.fac'),
100                           c('m.lo', 'm.hi.fac')] %*% bi.highjac)
101
102         uniskt.mean <- c(uniskt.mean, rawmean.uni.sd=drop(uni.sd))
103         lowskt.mean <- c(pii.low.sd=bi.piisd, lowskt.mean, rawmean.low.sd=drop(bi.lowsd))
104         highskt.mean <- c(highskt.mean, rawmean.high.sd=drop(bi.highsd))
105       }
106     }
107
108     # Reset the probability parameters of each bimodal distribution:
109     biskt <- fits$bimodal$skewt.pars
110
111     structure(c(unlist(uniskt), uniskt.mean, fits$unimodal$value,
112                 unlist(biskt[1,]), lowskt.mean,
113                 unlist(biskt[2,]), highskt.mean, fits$bimodal$value),
114               names=c(paste0(names(uniskt),'.uni'), names(uniskt.mean),
```

```
115                         'value.uni',
116                         paste0(colnames(biskt), '.low'), names(lowskt.mean),
117                         paste0(colnames(biskt), '.high'), names(highskt.mean),
118                         'value.bi'))
119     }
120
121     fitsum <- sapply(x, summarise.modality)
122     cat('\n') # Finish possible progress tracker
123     within(as.data.frame(t(fitsum)), valuediff <- value.bi-value.uni)
124 }
```

C

Fitting deterministic time-course models to the Tum−CI MFL data

This appendix describes how deterministic simulations of the MFL were implemented in R (Appendix C.1) and how they were then used to fit the model parameters to match either the LacZ assay (Appendix C.2) or flow cytometry (Appendix C.3 data sets.

C.1 Deterministic simulation of the MFL in R

All code presented in this appendix made extensive use of the `rootSolve`, `deSolve` [Soetaert et al., 2010] and `FME` [Soetaert and Petzoldt, 2010] R packages which provide general routines for finding steady-state roots of differential equations, simulating differential equations and optimising model parameters respectively. These libraries are loaded first.

```
1  ######################################################################
2  # Fit the ODE models of the MFL and controls to experimental data #
3  ######################################################################
4
5  library(rootSolve)
6  library(deSolve)
7  library(FME)
8
9  # Enable calculation of confidence intervals on modFit objects from the FME
10 # package:
11 vcov.modFit <- function(x,...) summary(x)$cov.scaled
```

With these libraries loaded, efficient routines for solving the MFL (`MFLderivs`) and Tum$^-$ control (`RepressDerivs`) ODEs were defined. These routines perform calculations in parallel for multiple values of the final level of P_{lac} induction (`PcFinal`). Routines are defined for finding both steady-state and time-course solutions to the ODEs, the latter further benefitting (in terms of speed) from an explicit definition of the Jacobian.

```
13
14 ### Define MFL ODEs and solvers for processing multiple Pc values in parallel ###
15 #{{{
```

```
16
17 ## tumciEquilibrate(Ct,Tt,params=MFLparams) ##
18 # Tum-CI equilibrium solver for initial values #
19 #{{{
20 tumciEquilibrate <- function(Ct,Tt,params=MFLparams){
21   RootEquation <- function(S,params)
22     with(params, S*(Es/(Tt-2*S))^Hs - Ct + S)
23   S <- uniroot(RootEquation, c(0,min(Ct,Tt/2)), params=params)$root
24   Cf <- Ct - S
25   Tf <- Tt - 2*S
26   list(S=S,Cf=Cf,Tf=Tf)
27 }
28 #}}}
29
30 ## steadyRepress(PcInit, PcFinal, params) ##
31 # Solve the Repressor ODEs at steady-state; safe for multiple PcFinal's
32 #{{{
33 steadyRepress <- function(PcInit,PcFinal,params)
34 {
35   initCf <- PcFinal/params[['Dc']]
36
37   # In this case, the initial values are exact:
38   return(data.frame(Pc=PcFinal, Cf=initCf,
39     Z=with(as.list(params), Pz/Dz/(1+(initCf/Er)^Hr))))
40
41   # The number of induction levels to solve for:
42   N <- length(PcFinal)
43
44   # Approximate the steady-state values of initial parameters
45   initCf <- PcInit/params[['Dc']]
46   initZ <- with(as.list(params), Pz/Dz/(1+(initCf/Er)^Hr))
47
48   # Initialise start values
49   init <- c(rep(initCf,N), rep(initZ,N))
50
51   derivs <- function(t, y, params, Pc, N){
52     # [Cf] = y[1], [Z] = y[N+1:]
53     Cf <- y[1:N]
54     Z <- y[(N+1):(2*N)]
55     with(as.list(params),
56         list(c(Pc - Dc*Cf,
57                 Pz/(1+(Cf/Er)^Hr) - Dz*Z)))
58   }
59
```

```
60    ss <- stode(init, parms=params, func=derivs, Pc=PcFinal, N=N)
61    with(ss, data.frame(Pc=PcFinal, Cf=y[1:N], Z=y[(N+1):(2*N)]))
62  }
63  #}}}
64
65  ## steadyMFL(PcInit,PcFinal,params) ##
66  # Solve the MFL ODEs at steady-state; safe for multiple PcFinal's
67  #{{{
68  steadyMFL <- function(PcInit,PcFinal,params)
69  {
70    # The number of induction levels to solve for:
71    N=length(PcFinal)
72
73    listparms <- as.list(params)
74
75    # Approximate the steady-state values of initial parameters
76    initCt <- PcInit/listparms$Dc
77    # Assume Tt = 'production due to repression by Ct'
78    steady <- tumciEquilibrate(initCt, with(listparms, Pt/Dt/(1+(initCt/Er)^Hr)),
79                               params=listparms)
80    # Improve the guess on Tt by one iteration:
81    steady <- tumciEquilibrate(initCt, with(c(listparms,steady), Pt/Dt/(1+(Cf/Er)^Hr)),
82                               params=listparms)
83    initZ <- with(c(listparms,steady), Pz/Dz/(1+(Cf/Er)^Hr))
84
85    # Initialise start values
86    init <- with(steady, c(rep(Cf,N), rep(Tf,N), rep(S,N), rep(initZ,N)))
87
88    derivs <- function(t, y, params, Pc, N){
89      # [C] = y[1], [T] = y[2], [S] = y[3], [Z]
90      Cf <- y[(0*N+1):(1*N)]
91      Tf <- y[(1*N+1):(2*N)]
92      S  <- y[(2*N+1):(3*N)]
93      Z  <- y[(3*N+1):(4*N)]
94      with(as.list(params),
95           list(c(Pc - Dc*Cf + ksoff*(-Cf*(Tf/Es)^Hs + S) + (Dt-Dil)*S,
96                  Pt/(1+(Cf/Er)^Hr) - Dt*Tf + 2*ksoff*(-Cf*(Tf/Es)^Hs + S)
97                  + (Dc-Dil)*S,
98                  -(Ds+Dc+Dt-Dil)*S + ksoff*(Cf*(Tf/Es)^Hs - S),
99                  Pz/(1+(Cf/Er)^Hr) - Dz*Z
100                 )))
101   }
102
103   ss <- stode(init, parms=params, func=derivs, Pc=PcFinal, N=N,
```

```
104                 positive=TRUE)
105                 # jactype='bandint',bandup=3,banddown=3)
106   with(ss, data.frame(Pc=PcFinal, Cf=y[(0*N+1):(1*N)], Tf=y[(1*N+1):(2*N)],
107                       S=y[(2*N+1):(3*N)], Z=y[(3*N+1):(4*N)]))
108 }
109 #}}}
110
111 ## RepressDerivs(t, y, params, Pc, N) ##
112 # Define the Repressor ODEs to process multiple Pc values in parallel. The
113 # variables for each Pc are grouped together in y so that the Jacobian is
114 # banded.
115 #{{{
116 RepressDerivs <- function(t, y, params, Pc, N){
117   # [C] = y[1], [Z] = y[2]
118   ymat <- matrix(y, nrow=2)
119   Cf <- ymat[1,]
120   Z  <- ymat[2,]
121
122   Rpow <- with(as.list(params), 1/(1+(Cf/Er)^Hr))
123
124   with(as.list(params),
125        list(as.vector(matrix(c(
126                 Pc - Dc*Cf,
127                 Pz*Rpow - Dz*Z
128               ),nrow=2,byrow=TRUE))))
129 }
130 #}}}
131
132 ## MFLderivs(t, y, params, Pc, N, full=FALSE) ##
133 # Define the MFL ODEs to process multiple Pc values in parallel. The variables
134 # for each Pc are grouped together in y so that the Jacobian is banded.
135 #{{{
136 MFLderivs <- function(t, y, params, Pc, N, full=FALSE){
137   # [C] = y[1], [T] = y[2], [S] = y[3], [Z] = y[4]
138   ymat <- matrix(y, nrow=4)
139   Cf <- ymat[1,]
140   Tf <- ymat[2,]
141   S  <- ymat[3,]
142   Z  <- ymat[4,]
143
144   Rpow <- with(as.list(params), 1/(1+(Cf/Er)^Hr))
145   Spow <- with(as.list(params), ksoff*Cf*(Tf/Es)^Hs)
146
147   with(as.list(params),
```

```
148         list(as.vector(matrix(c(
149                 Pc - Dc*Cf - Spow + (ksoff+Dt-Dil)*S,
150                 Pt*Rpow - Dt*Tf - 2*Spow + 2*(ksoff+Dc-Dil)*S,
151                 -(ksoff+Ds+Dc+Dt-Dil)*S + Spow,
152                 Pz*Rpow - Dz*Z
153               ),nrow=4,byrow=TRUE))))
154 }
155 #}}}
156
157 ## MFLjacob(t, y, params, Pc, N, full=FALSE) ##
158 # For improved accuracy, define a Jacobian for the above multi-Pc MFL ODEs. It
159 # can be returned either as a full or banded matrix. Note that if N=1, the
160 # banded version is larger than the full matrix, and the ODE solver no longer
161 # works with the banded version.
162 #{{{
163 MFLjacob <- function(t, y, params, Pc, N, full=FALSE){
164   # [C] = y[1], [T] = y[2], [S] = y[3], [Z] = y[4]
165   ymat <- matrix(y, nrow=4)
166   Cf <- ymat[1,]
167   Tf <- ymat[2,]
168   S  <- ymat[3,]
169   Z  <- ymat[4,]
170
171   # Jacobian layout:
172   # dC[1]dot/dC[1], dC[1]dot/dT[1], dC[1]dot/dS[1], dC[1]dot/dZ[1], dC[1]dot/dC[2], ...
173   # dT[1]dot/dC[1], dT[1]dot/dT[1], dT[1]dot/dS[1], dT[1]dot/dZ[1],...
174   # dS[1]dot/dC[1], dS[1]dot/dT[1], dS[1]dot/dS[1], dS[1]dot/dZ[1],...
175   # dZ[1]dot/dC[1], dZ[1]dot/dT[1], dZ[1]dot/dS[1], dZ[1]dot/dZ[1],...
176   # dC[2]dot/dC[1], dC[2]dot/dT[1], dC[2]dot/dS[1], dC[2]dot/dZ[1],...
177   #
178   # Banded Jacobian layout:
179   # 0             , 0             , 0             , dC[1]dot/dZ[1], 0
    , ...
180   # 0             , 0             , dC[1]dot/dS[1], dT[1]dot/dZ[1], 0
    , ...
181   # 0             , dC[1]dot/dT[1], dT[1]dot/dS[1], dS[1]dot/dZ[1], 0
    , ...
182   # dC[1]dot/dC[1], dT[1]dot/dT[1], dS[1]dot/dS[1], dZ[1]dot/dZ[1], dC[2]dot/dC[2], ...
183   # dT[1]dot/dC[1], dS[1]dot/dT[1], dZ[1]dot/dS[1], 0             , dT[2]dot/dC[2], ...
184   # dS[1]dot/dC[1], dZ[1]dot/dT[1], 0             , 0             , dS[2]dot/dC[2], ...
185   # dZ[1]dot/dC[1], 0             , 0             , 0             , dZ[2]dot/dC[2], ...
186
187   zeros <- rep(0,ncol(ymat))
188   zerorow <- matrix(zeros,nrow=1)
```

```
189
190   # Use the 'zeros' vector to ensure correct vector lengths when constructing
191   # the Jacobian matrices with zero and constant terms:
192   with(as.list(params),{
193         Rpow <- 1/(1+(Cf/Er)^Hr)
194         dRpow.dC <- -Rpow*Rpow*(Hr/Er)*(Cf/Er)^(Hr-1)
195         dSpow.dC <- ksoff*(Tf/Es)^Hs
196         dSpow.dT <- (Hs/Es)*ksoff*Cf*(Tf/Es)^(Hs-1)
197         ddC <- matrix(c(-Dc-dSpow.dC, # dCdt
198                         Pt*dRpow.dC - 2*dSpow.dC, # dTdt
199                         dSpow.dC, # dSdt
200                         Pz*dRpow.dC), # dZdt
201                       nrow=4, byrow=TRUE)
202         ddT <- matrix(c(-dSpow.dT, # dCdt
203                         -Dt - 2*dSpow.dT, # dTdt
204                         dSpow.dT, # dSdt
205                         zeros), # dZdt
206                       nrow=4, byrow=TRUE)
207         ddS <- matrix(c(ksoff+Dt-Dil + zeros, # dCdt
208                         2*(ksoff+Dc-Dil) + zeros, # dTdt
209                         -(ksoff+Ds+Dc+Dt-Dil) + zeros, # dSdt
210                         zeros), # dZdt
211                       nrow=4, byrow=TRUE)
212         ddZ <- matrix(c(zeros, # dCdt
213                         zeros, # dTdt
214                         zeros, # dSdt
215                         -Dz + zeros), # dZdt
216                       nrow=4, byrow=TRUE)
217
218         if(ncol(ymat)==1)
219           return(cbind(as.vector(ddC),as.vector(ddT),as.vector(ddS),as.vector(ddZ)))
220
221         if(full){
222           fullmat <- cbind(as.vector(ddC),as.vector(ddT),as.vector(ddS),as.vector(ddZ))
223           fullmat <- cbind(fullmat, matrix(0, nrow=length(y), ncol=length(y)-4))
224           fullmat <- matrix(t(fullmat),nrow=4*length(y),ncol=ncol(ymat))
225           fullmat <- rbind(fullmat, matrix(0, nrow=4, ncol=ncol(ymat)))
226           fullmat <- matrix(as.vector(fullmat)[1:(length(fullmat)-length(y))],
227                             nrow=length(y),ncol=length(y),byrow=TRUE)
228           return(fullmat)
229         }
230         else{
231           bandmat <-
232             rbind(zerorow, zerorow, zerorow, ddC,
```

```
233                     zerorow, zerorow, ddT, zerorow,
234                     zerorow, ddS, zerorow, zerorow,
235                     ddZ, zerorow, zerorow, zerorow)
236             return(matrix(bandmat,nrow=7))
237           }
238           })
239 }
240 #}}}
241
242 ## detRepress(PcInit, PcFinal, params, assaytimes) ##
243 # Run a multi-Pc deterministic Repressor simulation starting from a single
244 # initial condition (PcInit). The simulation starts from the steady-state at
245 # PcInit (trivial in the case of the Repressor ODEs).
246 #{{{
247 detRepress <- function(PcInit, PcFinal, params, assaytimes)
248 {
249   # The number of induction levels to solve for:
250   N=length(PcFinal)
251
252   listparms <- as.list(params)
253
254   # Calculate the steady-state values of initial parameters
255   initCf <- PcInit/listparms$Dc
256   initZ <- with(listparms, Pz/Dz/(1+(initCf/Er)^Hr))
257   init <- rep(c(initCf, initZ), N)
258   # Above is exact, so no need to equilibrate
259
260   Ntimesteps <- 10
261
262   # Scale timesteps to the equivalent step size for 'assaytimes'
263   if(length(assaytimes)>1){
264     assaytimes <- assaytimes[order(assaytimes)]
265     assaytimeindex <- (round(assaytimes/max(assaytimes)*Ntimesteps)
266                        + 1:(length(assaytimes)))
267     Ntimesteps <- c(assaytimeindex[1]-1,diff(assaytimeindex))
268     timesteps <- mapply(function(x,y,z) seq(x,y,length.out=z),
269                         c(0,assaytimes[-length(assaytimes)]),
270                         assaytimes, Ntimesteps+1)
271     timesteps <- Reduce(function(x,y) c(x,y[-1]), timesteps, init=0)
272   } else {
273     timesteps <-
274       seq(0,assaytimes,length.out=Ntimesteps)
275   }
276
```

```
277   atol <- 1e-6
278   rtol <- 1e-6
279
280   # Simulate over the assay time starting from equilibration conditions:
281   repeat {
282     sim <- lsoda(init, times=timesteps, parms=params,
283                  func=RepressDerivs, atol=atol, rtol=rtol,
284                  Pc=PcFinal, N=N)
285
286     # Stop looping if all values are ok:
287     if(!any(is.na(sim))) break
288
289     if(length(assaytimes)>1){
290       assaytimeindex <- assaytimeindex*5
291       Ntimesteps <- c(assaytimeindex[1]-1,diff(assaytimeindex))
292       timesteps <- mapply(function(x,y,z) seq(x,y,length.out=z),
293                           c(0,assaytimes[-length(assaytimes)]),
294                           assaytimes, Ntimesteps+1)
295       timesteps <- Reduce(function(x,y) c(x,y[-1]), timesteps, init=0)
296     } else {
297       Ntimesteps <- 5*Ntimesteps
298       timesteps <-
299         seq(0,assaytimes,length.out=Ntimesteps)
300     }
301
302     atol <- atol*1e-2
303     rtol <- rtol*1e-2
304     if(atol<1e-10 || rtol<1e-10){
305       warning('NA values in repressor simulation with params:\n')
306       # print(as.list(params))
307       browser()
308       # Simulation failed, return zero instead of NA:
309       sim <- matrix(0,nrow=1,ncol=2*N+1)
310       break
311     }
312   }
313
314   if(length(assaytimes)>1){
315     finaldata <- data.frame(Pc=PcFinal)
316     for(i in 1:length(assaytimeindex)){
317       imat <- matrix(sim[assaytimeindex[i],-1], nrow=2)
318       idata <- data.frame(Cf=imat[1,], Z=imat[2,])
319       colnames(idata) <- paste0('t',i,colnames(idata))
320       finaldata <- cbind(finaldata, idata)
```

```
321       }
322       return(finaldata)
323     } else {
324       finalmat <- matrix(sim[nrow(sim),-1], nrow=2)
325       return(data.frame(Pc=PcFinal, Cf=finalmat[1,], Z=finalmat[2,]))
326     }
327   }
328   #}}}
329
330   ## detMFL(PcInit, PcFinal, params, assaytimes, ontime=60) ##
331   # Run a multi-Pc deterministic MFL simulation starting from a single initial
332   # condition (PcInit). The simulation starts by approximating steady-state
333   # values at PcInit and then equilibrating at the initial condition for
334   # 'ontime' minutes. It then solves the MFL ODEs for each 'PcFinal' value to
335   # the maximum time in 'assaytimes', saving the state at each time specified in
336   # 'assaytimes'
337   #{{{
338   detMFL <- function(PcInit, PcFinal, params, assaytimes, ontime=60)
339   {
340     # The number of induction levels to solve for:
341     N <- length(PcFinal)
342
343     listparms <- as.list(params)
344
345     # Allow the use of an optional scaling parameter for Tum production in the
346     # O/N cultures:
347     if(is.null(listparms[['PtInitScale']]))
348       listparms[['PtInitScale']] <- 1
349
350     # Approximate the steady-state values of initial parameters
351     initCt <- PcInit/listparms$Dc
352     # Assume Tt = 'production due to repression by Ct'
353     steady <-
354       tumciEquilibrate(initCt, with(listparms,
355                                     PtInitScale*Pt/Dt/(1+(initCt/Er)^Hr)),
356                        params=listparms)
357     # Improve the guess on Tt by one iteration:
358     steady <-
359       tumciEquilibrate(initCt, with(c(listparms,steady),
360                                     PtInitScale*Pt/Dt/(1+(Cf/Er)^Hr)),
361                        params=listparms)
362     initZ <- with(c(listparms,steady), Pz/Dz/(1+(Cf/Er)^Hr))
363
364     # Equilibrate at initial condition:
```

```
365   init <- with(steady, c(Cf=Cf, Tf=Tf, S=S, Z=initZ))
366
367   if(any(is.na(init))){
368     warning('Approximated initial steadystate values contain NA')
369     browser() # If the equilibration failed switch to allow user input
370   }
371
372   initparams <- params
373   initparams[['Pt']] <- with(listparms, PtInitScale*Pt)
374
375   Ntimesteps <- 10
376   maxatol <- 1e-6
377   maxrtol <- 1e-6
378   rtol <- c(rep(maxrtol,3), 1e-6)
379   atol <- c(rep(maxatol,3), 1e-6)
380
381   repeat{
382     sim <- lsoda(init, times=seq(0,ontime,length.out=Ntimesteps), parms=initparams,
383                  jacfunc=MFLjacob, jactype='fullusr',
384                  func=MFLderivs, atol=atol, rtol=rtol,
385                  Pc=PcInit, N=1, full=TRUE)
386
387     # Stop looping if all values are ok:
388     if(!any(is.na(sim))) break
389
390     Ntimesteps <- Ntimesteps*5
391     maxatol <- maxatol*1e-2
392     maxrtol <- maxrtol*1e-2
393     if(maxatol<1e-10 || maxrtol<1e-10){
394       warning('NA values in simulated initial conditions with params:\n')
395       # print(as.list(params))
396       bestcol <- 1
397       # browser() # disable browser...
398       # Use the best guess we have:
399       sim <- t(sim[bestcol,]) # Defaults to initial values
400       break
401     }
402     rtol <- c(rep(maxrtol,3), 1e-6)
403     atol <- c(rep(maxatol,3), 1e-6)
404   }
405
406   ### NB: A banded Jacobian can only be specified for N>1!!!! ###
407   ### Otherwise a *full* matrix must be used.
408
```

```
409  # Initialise start values from equilibrated values
410  init <- with(as.list(sim[nrow(sim),]), rep(c(Cf, Tf, S, Z), N))
411
412  # Scale timesteps to the equivalent step size for 'assaytimes'
413  if(length(assaytimes)>1){
414    assaytimes <- assaytimes[order(assaytimes)]
415    assaytimeindex <- (round(assaytimes/max(assaytimes)*Ntimesteps)
416                       + 1:(length(assaytimes)))
417    Ntimesteps <- c(assaytimeindex[1]-1,diff(assaytimeindex))
418    timesteps <- mapply(function(x,y,z) seq(x,y,length.out=z),
419                        c(0,assaytimes[-length(assaytimes)]),
420                        assaytimes, Ntimesteps+1)
421    timesteps <- Reduce(function(x,y) c(x,y[-1]), timesteps, init=0)
422  } else {
423    timesteps <-
424      seq(0,assaytimes,length.out=ceiling(assaytimes/ontime*Ntimesteps))
425  }
426
427  rtol <- rep(c(rep(maxrtol,3), 1e-6), N)
428  atol <- rep(c(rep(maxatol,3), 1e-6), N)
429
430  # Simulate over the assay time starting from equilibration conditions:
431  ### NB: A banded Jacobian can only be specified for N>1!!!! ###
432  ### Otherwise a *full* matrix must be used.
433  repeat {
434    if(N>1)
435      sim <- lsoda(init, times=timesteps, parms=params,
436                   jacfunc=MFLjacob, jactype='bandusr', bandup=3, banddown=3,
437                   func=MFLderivs, atol=atol, rtol=rtol,
438                   Pc=PcFinal, N=N, full=FALSE)
439    else
440      sim <- lsoda(init, times=timesteps, parms=params,
441                   jacfunc=MFLjacob, jactype='fullusr',
442                   func=MFLderivs, atol=atol, rtol=rtol,
443                   Pc=PcFinal, N=N, full=TRUE)
444
445    # Stop looping if all values are ok:
446    if(!any(is.na(sim))) break
447
448    if(length(assaytimes)>1){
449      assaytimeindex <- assaytimeindex*5
450      Ntimesteps <- c(assaytimeindex[1]-1,diff(assaytimeindex))
451      timesteps <- mapply(function(x,y,z) seq(x,y,length.out=z),
452                          c(0,assaytimes[-length(assaytimes)]),
```

```
453                             assaytimes, Ntimesteps+1)
454         timesteps <- Reduce(function(x,y) c(x,y[-1]), timesteps, init=0)
455       } else {
456         Ntimesteps <- 5*Ntimesteps
457         timesteps <-
458           seq(0,assaytimes,length.out=ceiling(assaytimes/ontime*Ntimesteps))
459       }
460
461       maxatol <- maxatol*1e-2
462       maxrtol <- maxrtol*1e-2
463       if(maxatol<1e-10 || maxrtol<1e-10){
464         warning('NA values in simulation with params:\n')
465         # print(as.list(params))
466         # Simulation failed, return zero instead of NA:
467         sim <- matrix(0,nrow=1,ncol=4*N+1)
468         break
469       }
470       rtol <- rep(c(rep(maxrtol,3), 1e-6), N)
471       atol <- rep(c(rep(maxatol,3), 1e-6), N)
472     }
473
474     if(length(assaytimes)>1){
475       finaldata <- data.frame(Pc=PcFinal)
476       # If any rows contain NA values, or the simulation was terminated early,
477       # then simply report the last available non-NA value:
478       firstNA <- match(TRUE, apply(sim, 1, function(x) any(is.na(x))),
479                        nomatch=nrow(sim)+1)
480       assaytimeindex[assaytimeindex>=firstNA | assaytimeindex>nrow(sim)] <- firstNA-1
481       for(i in 1:length(assaytimeindex)){
482         imat <- matrix(sim[assaytimeindex[i],-1], nrow=4)
483         idata <- data.frame(Cf=imat[1,], Tf=imat[2,], S=imat[3,], Z=imat[4,])
484         colnames(idata) <- paste0('t',i,colnames(idata))
485         finaldata <- cbind(finaldata, idata)
486       }
487       return(finaldata)
488     } else {
489       finalmat <- matrix(sim[nrow(sim),-1], nrow=4)
490       return(data.frame(Pc=PcFinal, Cf=finalmat[1,], Tf=finalmat[2,],
491                         S=finalmat[3,], Z=finalmat[4,]))
492     }
493   }
494   #}}}
495
496   #}}}
```

C.2 Fitting the model to the LacZ assays

The LacZ assay data required a number of steps including curation of the data set (Appendix C.2.1), setting up the model output (Appendix C.2.2), and defining the cost function for the optimisation routines (Appendix C.2.3).

C.2.1 Loading and curating the data set

The first step in fitting the LacZ assay data was to load and curate the data into a form that could be used with the `FME` package. The following code excerpt shows how this was done.

```
499  ### Load and process the data sets for fitting ###
500  #{{{
501  load('~/Documents/Biochemistry/Assays/pZCMFL/DRassays/CombinedAnalysis/MFLdata.Rdata')
502
503  sum(is.na(MFLdata$values)) # 19 NA values
504  sum(na.omit(MFLdata$values+1)<0) # 23 negative values
505
506  # AddValueMeans adds in the respective means and SDs to the data array after
507  # transformation by the 'transform' function
508  AddValueMeans <- function(df, name='id', transform=identity, ...){
509    valtransform <- transform(as.numeric(df$values), ...)
510    means <- with(df, tapply(valtransform, equil.strain.assay.on.iptg,
511                             mean, na.rm=TRUE))
512    sds <- with(df, tapply(valtransform, equil.strain.assay.on.iptg,
513                           sd, na.rm=TRUE))
514    Ns <- with(df, tapply(valtransform, equil.strain.assay.on.iptg,
515                          function(x) sum(!is.na(x))))
516    newcols <- paste0(name, c('.mean', '.sd', '.n', '.resid', '.scresid',
517                              '.sqscresid', '.seresid'))
518    df[[newcols[1]]] <- as.numeric(df$values)
519    df[[newcols[2]]] <- as.numeric(df$values)
520    for(i in names(means))
521      df[i==df$equil.strain.assay.on.iptg, newcols[1]] <- means[[i]]
522    for(i in names(sds))
523      df[i==df$equil.strain.assay.on.iptg, newcols[2]] <- sds[[i]]
524    for(i in names(Ns))
525      df[i==df$equil.strain.assay.on.iptg, newcols[3]] <- Ns[[i]]
526    # transform.resid (residuals):
527    df[,newcols[4]] <- valtransform - df[,newcols[1]]
528    # transform.scresid (residuals divided by standard deviation):
529    df[,newcols[5]] <- df[,newcols[4]]/df[,newcols[2]]
530    # transform.sqscresid (residuals divided by sqrt of standard deviation):
531    df[,newcols[6]] <- df[,newcols[4]]/sqrt(df[,newcols[2]])
```

```
532     # transform.seresid (residuals divided by standard error):
533     df[,newcols[7]] <- df[,newcols[4]]*sqrt(df[,newcols[3]])/df[,newcols[2]]
534     df
535   }
536
537   # Limit the analysis just to the ermc and wrmc strains
538   MFLdata.mc <- with(MFLdata, MFLdata[strain%in%c('ermc','wrmc'),])
539   # Reset the columns:
540   MFLdata.mc <- MFLdata.mc[,c("values","assay","on","iptg","strain","day",
541                               "assay.on.iptg.strain","equil","equil.strain.on",
542                               "assay.on","strain.assay.on",
543                               "equil.strain.assay.on.iptg","normiptg")]
544   default(MFLdata.mc$values) <- 'equil.strain.assay.on.iptg'
545
546   # Split the data into assay type:
547   MFLdata.summaries <-
548     split(MFLdata.mc$values, factor(getmask(MFLdata$values, '#assay+on')))
549   MFLdata.summaries <- lapply(MFLdata.summaries, summary)
550
551   # There don't seem to be any particular trends with assay type:
552   with(MFLdata.summaries,
553       matplot(log(matrix(c(TM.LO$mean,TM.HO$mean,TP.LO$mean,TP.HO$mean),ncol=4)),
554               log(matrix(c(TM.LO$sd,TM.HO$sd,TP.LO$sd,TP.HO$sd),ncol=4)),
555               type='p', pch=19, cex=0.5, col=c('black','grey','blue','red'),
556               xlab='Mean values', ylab='Standard deviation'))
557
558   ## Calculate data set statistics ##
559   MFLdata.mc <- AddValueMeans(MFLdata.mc)
560
561   MFLdata.mc <-
562     within(MFLdata.mc,{
563             slope <- coef(MFLdata.mc.sdfit)[[2]]
564             intercept <- coef(MFLdata.mc.sdfit)[[1]]
565             sd.est <- ((id.mean+1)^slope)*exp(intercept)
566             sdtrend.sc <- id.resid/sd.est
567         })
568
569   ##############################################################################
570   ## Normalisation Transformation Summary ##
571   # No additional transformation of the data is necessary, but sample means
572   # should be weighted by the fitted standard deviation and not individual
573   # sample SDs.
574   ##############################################################################
575
```

```
## Build the set of Pc values used in the experiments ##
#{{{
# To use the same simulation data for assays from both equilibration times,
# a few classification aids need to be used. Firstly, since
# the normal and long equilibration time assays used slightly different sets
# of final IPTG concentrations, to avoid repeating the same simulations for
# the same IPTG concentrations, we identify Pc values by a unique ID
Ival <- with(MFLdata, split(as.numeric(as.character(iptg)),equil))
Ival <- lapply(Ival,unique)
Ival <- lapply(Ival, function(x) x[order(x)])
with(Ival, intersect(normal,long)) # 0  20  40  80 120 300
with(Ival, setdiff(normal,long)) # 160 220
IvalSet <- with(Ival, union(normal,long))
IvalSet <- IvalSet[order(IvalSet)]
with(Ival, which(IvalSet %in% intersect(normal,long))) # 1  2  3  5  6 10
which(IvalSet %in% Ival$normal) # 1  2  3  5  6  7  9 10
which(IvalSet %in% Ival$long) # 1  2  3  4  5  6  8 10

# Use the following masks on ordered 'IvalSets' to index Pc values for
# normal or long equilibration experiments:
normalPcMask <- c(1:3,5:7,9:10) # Copied from set analysis above
longPcMask <- c(1:6,8,10) # Copied from set analysis above

# Construct lists of all the different possible Pc values depending on
# strain (ermc,...), assay (Tum+/-) and overnight (high/low):
ExptPcValues <- with(MFLdata, split(normiptg, paste0(strain,assay,on)))
ExptPcValues <- lapply(ExptPcValues, unique)
ExptPcValues <- lapply(ExptPcValues, function(x) x[order(x)])

# Each item in ExptPcValues can be indexed by normal or long PcMask variants
# to obtain the values for either of those experiments.
#}}}

# Obtain the initial conditions for each assay variant:
initPc <- with(MFLdata,
               list(Pc.WPL=min(normiptg[assay.on=='TP.LO'&strain=='wrmc']),
                    Pc.WPH=max(normiptg[assay.on=='TP.HO'&strain=='wrmc']),
                    Pc.WML=min(normiptg[assay.on=='TM.LO'&strain=='wrmc']),
                    Pc.WMH=max(normiptg[assay.on=='TM.HO'&strain=='wrmc']),
                    Pc.EPL=min(normiptg[assay.on=='TP.LO'&strain=='ermc']),
                    Pc.EPH=max(normiptg[assay.on=='TP.HO'&strain=='ermc']),
                    Pc.EML=min(normiptg[assay.on=='TM.LO'&strain=='ermc']),
                    Pc.EMH=max(normiptg[assay.on=='TM.HO'&strain=='ermc'])))

```

```
620   # Create a summary data array for fitting {{{
621   MFLsummary <-
622     with(within(MFLdata, {
623                 values <- as.numeric(values)
624                 grouping <- equil.strain.assay.on.iptg
625                 iptg <- as.numeric(as.character(iptg))}),
626           data.frame(name='name', # Temporary placeholder to be filled later
627                      PcID=1, # Temporary placeholder to be filled later
628                      Zmean=tapply(values, grouping, mean, na.rm=TRUE),
629                      Zsd=tapply(values, grouping, sd, na.rm=TRUE),
630                      iptg=tapply(iptg, grouping, head, n=1),
631                      Pc=tapply(normiptg, grouping, head, n=1),
632                      equil=tapply(as.character(equil), grouping, head, n=1),
633                      strain=tapply(as.character(strain), grouping, head, n=1),
634                      assay=tapply(as.character(assay), grouping, head, n=1),
635                      on=tapply(as.character(on), grouping, head, n=1)
636                      ))
637
638   # Replace sample SDs with fitted SDs:
639   MFLsummary$Zsd <-
640     with(list(sd.slope=coef(MFLdata.mc.sdfit)[[2]],
641               sd.intercept=coef(MFLdata.mc.sdfit)[[1]]),
642          ((MFLsummary$Zmean+1)^sd.slope)*exp(sd.intercept))
643
644   # Define a name identifier to be used when comparing with modelling output
645   MFLsummary <-
646     within(MFLsummary, {
647             name <- paste(strain, assay, on, sep='.')
648             # Name as t1Z for normal or t2Z for long equilibration times. The
649             # 'Z' selects that simulation variable in particular.
650             name[equil=='normal'] <- paste0(name[equil=='normal'],'.t1Z')
651             name[equil=='long'] <- paste0(name[equil=='long'],'.t2Z')
652           })
653
654   # Order elements first by group name, then by IPTG. This needs to be done
655   # before allocating a PcID so that the ordering is correct.
656   MFLsummary <- MFLsummary[with(MFLsummary,order(as.character(name),iptg)),]
657   # Use PcIDs to identify each IPTG factor based on the Pc value set identity
658   # allocated above:
659   MFLsummary <-
660     within(MFLsummary, {
661             PcID[equil=='normal'] <-
662               rep(normalPcMask, length.out=length(PcID[equil=='normal']))
663             PcID[equil=='long'] <-
```

```
664                 rep(longPcMask, length.out=length(PcID[equil=='long']))
665             })
666     #}}}
667
668     #}}}
```

C.2.2 Setting up the model output

Once the data was loaded, simulations of the deterministic model needed to be configured to output simulations only for the experimental induction levels. The output of these simulations also needed to be in a form that would be recognised by the FME package.

```
670     ### Set up the modelling environment ###
671     #{{{
672     ## Create a new parameter set for MFL fitting ##
673     # Start with:
674     # - rbsScale=1 (i.e., assuming correct experimental parameters)
675     # - LOscale=HOscale=1 (assuming correct initial Pc values)
676     # - Take all other required parameters from the experimentally derived set
677     assaytimes.expt <- c(5.9,7.5)*60 # not specified in paramsExpt
678     FitPars.init <- c(unlist(paramsExpt[c('Es','Hs','Pt','Dt','Ds','Dil','ksoff',
679                                           'Er','Hr','Pz','Dz','Dc')]),
680                       rbsScale=1, LOscale=1, HOscale=1, toff=assaytimes.expt[[1]])
681
682     ### Define models as functions of Ct ###
683     #{{{
684
685     # Define initial and experimental Pc values as Ct values:
686     ExptCtValues.ermc <- lapply(ExptPcValues[grep('^ermc',names(ExptPcValues))],
687                                 function(x) x/paramsExpt$Dc)
688     ExptCtValues.wrmc <- lapply(ExptPcValues[grep('^wrmc',names(ExptPcValues))],
689                                 function(x) x/paramsExpt$Dc)
690     ExptCtInit.ermc <- lapply(initPc[grep('^Pc\\.E',names(initPc))],
691                               function(x) x/paramsExpt$Dc)
692     ExptCtInit.wrmc <- lapply(initPc[grep('^Pc\\.W',names(initPc))],
693                               function(x) x/paramsExpt$Dc)
694     LOermc.mask <- grepl('L$', names(ExptCtInit.ermc))
695     HOermc.mask <- grepl('H$', names(ExptCtInit.ermc))
696     LOwrmc.mask <- grepl('L$', names(ExptCtInit.wrmc))
697     HOwrmc.mask <- grepl('H$', names(ExptCtInit.wrmc))
698
699     ## Separate model functions ##
700     MFLmodel.Ct.wrmc <- function(params, assaytimes=c(5.9*60, 7.5*60)){
701       initPc.wrmc <- with(as.list(params), unlist(ExptCtInit.wrmc)*Dc)
702       initPc.wrmc[LOwrmc.mask] <- initPc.wrmc[LOwrmc.mask]*params['LOscale']]
```

```
703     initPc.wrmc[HOwrmc.mask] <- initPc.wrmc[HOwrmc.mask]*params[['HOscale']]
704     PcValues.wrmc <-
705       with(as.list(params), lapply(ExptCtValues.wrmc, function(x) x*Dc))
706     with(c(as.list(initPc.wrmc), PcValues.wrmc),
707          data.frame(PcID=1:10,
708                     wrmc.TP.LO=detMFL(Pc.WPL,wrmcTPLO,params,assaytimes),
709                     wrmc.TP.HO=detMFL(Pc.WPH,wrmcTPHO,params,assaytimes),
710                     wrmc.TM.LO=detRepress(Pc.WML,wrmcTMLO,params,assaytimes),
711                     wrmc.TM.HO=detRepress(Pc.WMH,wrmcTMHO,params,assaytimes)))
712   }
713
714   MFLmodel.Ct.ermc <- function(params, assaytimes=c(5.9*60, 7.5*60)){
715     initPc.ermc <- with(as.list(params), unlist(ExptCtInit.ermc)*rbsScale*Dc)
716     initPc.ermc[LOermc.mask] <- initPc.ermc[LOermc.mask]*params[['LOscale']]
717     initPc.ermc[HOermc.mask] <- initPc.ermc[HOermc.mask]*params[['HOscale']]
718     PcValues.ermc <-
719       with(as.list(params), lapply(ExptCtValues.ermc, function(x) x*rbsScale*Dc))
720     with(c(as.list(initPc.ermc), PcValues.ermc),
721          data.frame(PcID=1:10,
722                     ermc.TP.LO=detMFL(Pc.EPL,ermcTPLO,params,assaytimes),
723                     ermc.TP.HO=detMFL(Pc.EPH,ermcTPHO,params,assaytimes),
724                     ermc.TM.LO=detRepress(Pc.EML,ermcTMLO,params,assaytimes),
725                     ermc.TM.HO=detRepress(Pc.EMH,ermcTMHO,params,assaytimes)))
726   }
727
728   ## Combined model function ##
729   MFLmodel.Ct <- function(params, assaytimes=c(5.9*60, 7.5*60)){
730     initPc.wrmc <- with(as.list(params), unlist(ExptCtInit.wrmc)*Dc)
731     initPc.wrmc[LOwrmc.mask] <- initPc.wrmc[LOwrmc.mask]*params[['LOscale']]
732     initPc.wrmc[HOwrmc.mask] <- initPc.wrmc[HOwrmc.mask]*params[['HOscale']]
733     PcValues.wrmc <-
734       with(as.list(params), lapply(ExptCtValues.wrmc, function(x) x*Dc))
735
736     initPc.ermc <- with(as.list(params), unlist(ExptCtInit.ermc)*rbsScale*Dc)
737     initPc.ermc[LOermc.mask] <- initPc.ermc[LOermc.mask]*params[['LOscale']]
738     initPc.ermc[HOermc.mask] <- initPc.ermc[HOermc.mask]*params[['HOscale']]
739     PcValues.ermc <-
740       with(as.list(params), lapply(ExptCtValues.ermc, function(x) x*rbsScale*Dc))
741
742     with(c(as.list(initPc.wrmc), as.list(initPc.ermc), PcValues.wrmc, PcValues.ermc),
743          data.frame(PcID=1:10,
744                     wrmc.TP.LO=detMFL(Pc.WPL,wrmcTPLO,params,assaytimes),
745                     wrmc.TP.HO=detMFL(Pc.WPH,wrmcTPHO,params,assaytimes),
746                     wrmc.TM.LO=detRepress(Pc.WML,wrmcTMLO,params,assaytimes),
```

```
                    wrmc.TM.HO=detRepress(Pc.WMH,wrmcTMHO,params,assaytimes),
                    ermc.TP.LO=detMFL(Pc.EPL,ermcTPLO,params,assaytimes),
                    ermc.TP.HO=detMFL(Pc.EPH,ermcTPHO,params,assaytimes),
                    ermc.TM.LO=detRepress(Pc.EML,ermcTMLO,params,assaytimes),
                    ermc.TM.HO=detRepress(Pc.EMH,ermcTMHO,params,assaytimes)))
}

## Test the new functions ##
MFLmodel.Ct(FitPars.init)
MFLmodel.Ct.wrmc(FitPars.init)
MFLmodel.Ct.ermc(FitPars.init)

## Function for plotting curve details rather than just data points ##
MFLmodel.Ct.curves <- function(params, assaytime=7.5*60, n=200){
  # Obtain the assay limits for determining curve points:
  initPc.wrmc <- with(as.list(params), unlist(ExptCtInit.wrmc)*Dc)
  initPc.ermc <- with(as.list(params), unlist(ExptCtInit.ermc)*rbsScale*Dc)

  # Calculate the simulation Pc's:
  simPc <- with(as.list(c(initPc.wrmc,initPc.ermc)),
                list(ermcTP=seq(Pc.EPL,Pc.EPH,length.out=n),
                     ermcTM=seq(Pc.EML,Pc.EMH,length.out=n),
                     wrmcTP=seq(Pc.WPL,Pc.WPH,length.out=n),
                     wrmcTM=seq(Pc.WML,Pc.WMH,length.out=n)))

  # Now scale the initial Pc states:
  initPc.wrmc[LOwrmc.mask] <- initPc.wrmc[LOwrmc.mask]*params['LOscale']]
  initPc.wrmc[HOwrmc.mask] <- initPc.wrmc[HOwrmc.mask]*params['HOscale']]
  initPc.ermc[LOermc.mask] <- initPc.ermc[LOermc.mask]*params['LOscale']]
  initPc.ermc[HOermc.mask] <- initPc.ermc[HOermc.mask]*params['HOscale']]

  # Finally simulate:
  with(c(as.list(initPc.wrmc), as.list(initPc.ermc), simPc),
       data.frame(ermc.TP.LO=detMFL(Pc.EPL,ermcTP,params,assaytime),
                  ermc.TP.HO=detMFL(Pc.EPH,ermcTP,params,assaytime),
                  ermc.TM.LO=detRepress(Pc.EML,ermcTM,params,assaytime),
                  ermc.TM.HO=detRepress(Pc.EMH,ermcTM,params,assaytime),
                  wrmc.TP.LO=detMFL(Pc.WPL,wrmcTP,params,assaytime),
                  wrmc.TP.HO=detMFL(Pc.WPH,wrmcTP,params,assaytime),
                  wrmc.TM.LO=detRepress(Pc.WML,wrmcTM,params,assaytime),
                  wrmc.TM.HO=detRepress(Pc.WMH,wrmcTM,params,assaytime)))
}
#}}}
```

```
#}}}
```

C.2.3 Defining the cost function and optimising parameters

Finally, a cost function is defined by making use of the `modCost` function of the `FME` package. A selection of parameters are chosen to be free to vary, whilst others are set fixed. The model can then be fit to the data using the 'Pseudo' algorithm of the `modFit` function.

```
### Fit normal and long WR-MC and ER-MC data together, adjusting initial values ###
# Vary all parameters including the rate of dilution 'Dil' (hence also 'Dc'
# and 'Dz') and the scaling on initial low & high O/N values, but excluding
# the time offset 'toff', 'ksoff', 'Ds', and 'rbsScale'. The pseudo-random
# optimisation is started from the best fit of the normal and long WR-MC
# data.
#{{{
# Calculate initial parameters:
erwrl.initpars <- wrmcl.optpars
# Add in optimised 'rbsScale' value:
erwrl.initpars['rbsScale'] <- rbsScale.opt

erwrl.vary <-
  c('Er','Hr','Es','Hs','Mt','Mz','Dil','DtScale','LOscale','HOscale')
erwrl.derived <- c('Dc','Pt','Dt','Pz','Dz')
erwrl.fixed <-
  names(erwrl.initpars)[!names(erwrl.initpars)%in%c(erwrl.vary,erwrl.derived)]
erwrl.varpars <- erwrl.initpars[erwrl.vary]
erwrl.fixpars <- erwrl.initpars[erwrl.fixed]
erwrl.lpars <- log(erwrl.varpars)

erwrl.costlog <- function(lparams){
  # Update the parameters
  params <- c(exp(lparams), erwrl.fixpars)
  assaytimes <- params['toff']] + c(0,96) # Fixed time interval
  params[c('Dc','Dz')] <- params['Dil']]
  params['Dt']] <- with(as.list(params), Dil*DtScale)
  params['Pt']] <- with(as.list(params), Mt*Dt)
  params['Pz']] <- with(as.list(params), Mz*Dz)

  # Progress report:
  counter <- get('FitCounter', envir=.GlobalEnv)
  if(counter%%10 == 0){ cat('\n'); print(params); cat('\n') }
  assign('FitCounter', counter+1, envir=.GlobalEnv)
  cat(paste0(counter+1,'/10000, '))

  modCost(model=MFLmodel.Ct(params, assaytimes=assaytimes),
```

```
1597              obs=MFLsummary[MFLsummary$strain%in%c('wrmc','ermc'),],
1598              x='PcID', y='Zmean', err='Zsd')
1599  }
1600
1601  FitCounter <- 0
1602  erwrl.costlog(erwrl.lpars)
1603
1604  ## Define some limits on the variable parameters ##
1605  erwrl.lower <-
1606    c(Er= 100, Hr=1, Es= 500, Hs=1, Mt=6000, Mz=500, Dil=log(2)/40,
1607      DtScale=1, LOscale=0.1, HOscale=0.1)
1608  erwrl.upper <-
1609    c(Er=2400, Hr=3, Es=10000, Hs=5, Mt=32000, Mz=1300, Dil=log(2)/27,
1610      DtScale=20, LOscale=5, HOscale=5)
1611
1612  # Take logs of limits (and reorder the limits to match the parameters):
1613  erwrl.lower <- log(erwrl.lower[erwrl.vary])
1614  erwrl.upper <- log(erwrl.upper[erwrl.vary])
1615
1616  ### Fit the model using a Monte-Carlo approach ###
1617  FitCounter <- 0
1618  erwrl.fit <-
1619    modFit(f=erwrl.costlog, p=erwrl.lpars, method='Pseudo',
1620           lower=erwrl.lower, upper=erwrl.upper,
1621           control=list(verbose=TRUE))
1622
1623  ### Errors on the parameter estimates ###
1624  erwrl.fit.stderr <-
1625    data.frame(estimate=coef(erwrl.fit),
1626               lower=coef(erwrl.fit)-summary(erwrl.fit)$par[,'Std. Error'],
1627               upper=coef(erwrl.fit)+summary(erwrl.fit)$par[,'Std. Error'],
1628               conflow=confint(erwrl.fit)[,1], confup=confint(erwrl.fit)[,2])
1629  erwrl.fit.stderr <- exp(erwrl.fit.stderr)
1630  erwrl.fit.stderr <- within(erwrl.fit.stderr, lowerdev <- conflow-estimate)
1631  erwrl.fit.stderr <- within(erwrl.fit.stderr, upperdev <- confup-estimate)
1632  erwrl.fit.stderr <- within(erwrl.fit.stderr, meandev <- (abs(lowerdev)+abs(upperdev))/2)
1633
1634  #}}}
```

C.3 Fitting the model to the flow cytometry assays

Fits of the flow cytometry data set were performed in a simlar way to that described in the previous section for the LacZ assay data. However, the idiosyncracies of the flow

cytometry data set necessitated a number of amendments. These included (1) choosing appropriate 'deterministic' data points, (2) setting fluorescence intensity scaling and offset factors, (3) introducing additional model parameters, and (4) using an alternative optimisation algorithm. These amendments will be explained in the coming paragraphs.

The idiosyncracies of this new data necessitated a few modifications to the protocol, which are described in detail in Appendix C.3, but summarised in the following three points. (1) Unlike the activities obtained by LacZ assay, the fluorescence intensities measured by flow cytometry were not calibrated to an absolute reference. To relate fluorescence intensity to reporter gene activity, fluorescence scaling and offset parameters were chosen for each assay using preliminary alignments of the model and data. These per-assay parameters allow for systematic day-to-day variation in fluorescence measurements, essentially acting to calibrate each assay to a nominal reference value for the M_Z parameter (here, standing for the fluorescence produced from an unrepressed *pR* promoter). (2) There was insufficient flexibility in the initial model to simultaneously match the data at all equilibration time points, so three new parameters were introduced. An 'initial Tum scaling' parameter complemented the low and high O/N scaling parameters by effectively dissociating initial CI and Tum concentrations. A pair of 'time-offset' parameters, one for the low cell count assays and one for the high cell count assays, were used to permit a lag time before cells start log phase growth. (3) The greater apparent correspondence between the initial model predictions and the deterministic projection of the flow cytometry data motivated a directed approach to fitting the data instead of the pseudo-random search used to fit the LacZ assay data. It was more informative and efficient to incrementally optimise parameters, fitting related parameters and data sets together (e.g., those pertaining to the Tum^- control strains), whilst keeping other parameters constrained except for a final step of numerical optimisation with all parameters left unconstrained.

Without knowing *a priori* the likelihood of a sub-population belonging to the presumptive deterministic trajectory, it was assumed that the 'stochastic' sub-populations were those with intensities furthest from their initial states. This was a naïve choice of 'deterministic' data, but stood as a basic first guess to determine preliminary parameters. Since the deterministic model predicts a single macroscopic state, only one sub-population of each sample can be attributed to the deterministic trajectory. The choice of data to include was based on a heuristic assessment of sub-populations consistent with a deterministic interpretation. The assumption was that any sub-population centred in the 'alternative' zone of fluorescence (i.e., with intensities furthest from the initial state) arose stochastically by, for example, some low probability stochastic switching event shifting the state of the circuit into the basin of attraction of the alternative stable state. That is, the data was filtered to exclude low fluorescence points in low O/N samples and to exclude high fluorescence points in high O/N samples. So for bimodal low or high O/N samples, the expected sub-populations (i.e., those populations with fluorescence intensity closest to their initial

states) were kept and the alternative populations excluded. Unimodal low O/N samples with low fluorescence were also excluded to avoid the potential inclusion of dominant 'stochastic' populations that might mask smaller 'deterministic' populations. Similarly, unimodal samples with high fluorescence were discarded for the high O/N samples. An additional three points in the high O/N samples were also excluded since they diverged significantly from expected behaviour: these were small sub-populations in the normal equilibration time and high cell count long equilibration time assays whose locations fell substantially below the control curves at low CI production rates. All data excluded from the fits is indicated by the open circles in Figure C.1.

Unlike the activities obtained by LacZ assay, the fluorescence intensities measured by flow cytometry were not calibrated to an absolute reference, so additional care needed to be taken when relating these intensities to reporter gene activity. Fluorescence intensities measured by flow cytometry are sensitive functions of many instrumental settings (e.g., the laser strength and pulse width, the photomultiplier tube voltage, and the sample flow rate); these can be set consistently, but some level of day-to-day variation in fluorescence intensity remains unavoidable. The result is that relative fluorescence intensities within an assay can be reliably compared, but comparisons between assays are less reliable. Since cellular autofluorescence produces a non-zero background fluorescence, both this offset and the scale of fluorescence may vary across assays. To allow for this variation in the model, per-assay fluorescence offset and scaling parameters were added that could be optimised for a given assay. The scaling parameter is essentially equivalent to the M_Z parameter used when fitting the LacZ assay data. M_Z represents the maximum steady-state level of reporter protein that would be produced from the unrepressed *pR* promoter. Here, M_Z becomes the steady-state level of fluorescent protein produced from *pR* (in arbitrary fluorescence units instead of LacZ units), but to facilitate inter-assay comparison, it was set fixed to 1000, leaving the fluorescence scaling parameter to convert relative fluorescence intensities to this nominal absolute reference activity.

It was found that leaving the fluorescence offset and scaling parameters to freely vary within the fitting algorithm provided too much flexibility in the model: the optimisation function would frequently get trapped in poorly-fitting local minima. Hence, methods were sought that might objectively fix the fluorescence offset and scale for each assay. Fluorescence offset parameters were simple to determine, since all of the assays contained a number of data points that could be presumed to have close to zero activity. These data points were picked to be the low activity high O/N unimodal points in both the tum^+ and tum^- strains (an approximate filter for 'low *pR* activity' was obtained by restricting to data points at CI production rates greater than half of the maximum CI production rate). To allow for slight trends in the data, model curves simulated from a crude MFL parameter set were scaled and offset using linear regression to match the model curves to these selected low-activity data points. The fitted offsets were then kept as the fixed

fluorescence offset parameters for each assay.

Ideally, the fluorescence scaling parameters would then be fixed using high *pR* activity reference data points that are known to be consistent between assays. Unfortunately, such data points are difficult to identify in the present assays, since the high *pR* activity data points show fairly large (absolute) variation with equilibration time, which is then further compounded by greater experimental variability at the same activities. Instead, fluorescence scale parameters were chosen by performing flexible fits of the data for each assay separately. In these preliminary fits of the data, loose bounds were placed on all other model parameters, so that any flexibility that might be captured by these other parameters would not bias scaling parameter determination. In so doing, data points whose values are largely determined by the maximum production rate from *pR* (such as the high fluorescence low O/N data) also have the largest influence in determining the scaling parameter. For all subsequent fits of the data, the fluorescence scale parameter for each assay was set fixed to this optimised value.

Parameterisation of the model was much the same as was previously specified for fitting of the LacZ assays. As before, the core optimisation parameters are the EC_{50}s and Hill coefficients of the CI−*pR* and Tum−CI equilibria (ε_R, H_R, ε_S and H_S), the maximal steady-state Tum concentration (M_T), and the rates of protein loss (γ and δ_T). The only core parameter missing from this list is the maximal steady-state reporter activity (M_Z), since, as previously mentioned, it is set fixed to 1000 in favour of per-assay fluorescence scaling parameters. Recall that the rates of loss of CI and the reporter were both set by the dilution rate (γ). Further, the Tum degradation rate was also parameterised as a multiplier of γ, so that with a lower bound of one, δ_T would never be smaller (slower) than γ. The production rates necessary for simulation (p_C, p_T and p_Z) are then determined by the maximal steady-state concentrations and these rates of loss.

Induction levels of the P_{lac} promoter are set for each sample in terms of the calibrated steady-state concentrations of total CI. When fitting the LacZ assay data, a RBS scaling factor was applied to reduce the steady-state CI concentrations measured for production from the eRBS to those measured for production from the wRBS (see Section 4.3.3). Since it was likely that the eRBS measurements were an overestimate, this same scaling was applied when modelling the flow cytometry data. To confirm its applicability to the flow cytometry data set, the CI RBS scaling factor was applied to the ER-MC curve shown in Figure 5.9 for comparison with the WR-MC curve in that figure. The overlap of the two assays was as good as that previously observed for the LacZ assays. The steady-state CI concentrations remained fixed at these scaled values for all fits; CI production rates are, however, still able to vary since the dilution rate is allowed to vary.

The core parameters are complemented by three parameters whose roles are essentially to enable additional flexibility in the initial conditions. The introduction of such parameters is justified by the need to compensate for the anticipated differences between cultures

in log phase versus those in stationary phase (i.e., the initial O/N cultures). For fitting of the LacZ assays, low and high O/N scaling factors were introduced that could independently adjust the effective level of induction of the O/N cultures. This addition allowed for asymmetries in conditions between the two alternative O/N cultures and meant that steady-state CI concentrations in the O/N cultures could be greater or lesser than might be expected in a growing log-phase culture. Both of these parameters were retained for fitting the flow cytometry data. The O/N scaling parameters implicitly affect the initial concentration of Tum, which adopts steady-state values according to the adjusted induction levels. When fitting the short equilibration time flow cytometry assays, however, it became necessary to dissociate the initial Tum and CI concentrations so that the apparently slow evolution of the high fluorescence sub-populations might be matched. For this reason an additional parameter was introduced (identical for both O/N conditions) that would adjust the initial steady-state concentration of Tum by scaling the maximum Tum production rate (p_T).

A final pair of parameters were introduced to account for timing differences between the assay sets. In order to keep consistency with LacZ assay protocols, cultures were assayed by flow cytometry only when they reached an OD_{600} of 0.6. This meant that the actual time taken until cultures were assayed could vary between assays, even if the same equilibration protocol was followed. This is seen in Figure 5.11 when comparing the reported times to assay between the low and high cell count assay sets. The high cell count assays (lower panels) consistently take a shorter time to reach similar final OD_{600}s than the low cell count assays do, in spite of otherwise identical protocols. The time *intervals* between each equilibration method (i.e., about 1.4 hours between short and medium, and 1 hour between medium and long equilibration times) are, however, very similar between the sets. Thus the discrepancy appears to be differing lag times occuring early in the assay, perhaps pertaining to the time it takes to transition from stationary-phase to log-phase growth. The origin of these differences is unknown, but likely signifies some unaccounted systematic difference in the states of the O/N cultures. To account for this divergent lag time, two time offset parameters, one for each assay set, were introduced into the model. These time offset parameters reduce the effective times taken to reach the short equilibration time point, but do not change the relative times between equilibration assays within each set. The introduction of these parameters allowed for much better alignment of the model with the combined assay sets, particularly when modelling the Tum^- control data. It also dissociated fitting of the dilution rate and Tum degradation rate parameters from the presumptive initial conditions.

Parameter optimisation was performed incrementally, fitting the model to the flow cytometry data set in stages. This is in contrast with the all-parameter pseudo-random search that was used to find parameters for the LacZ assay data set. For the flow cytometry data, the relationship between model and data was more apparent, so a directed approach to

fitting the data became more efficient and informative. Thus, instead of allowing all parameters to freely vary, protocols were implemented to hold certain parameters fixed in early optimisation iterations, but with progressively more parameters allowed to vary as the model better matched the data. Numerical optimisation of the unconstrained parameters in each iteration was made using a quasi-Newton method of the R `optim` function that performs a directed search of parameter space based on function values and gradients. In final optimisation iterations, all parameters in this numerical optimisation algorithm were left unconstrained.

After preliminary parameter optimisation for setting the per-assay fluorescence offset and scale parameters, all other model parameters were subsequently optimised to simultaneously fit all six assays. In keeping with the directed fitting approach, the data for the Tum^- control strains was fit first to establish tighter constraints on the subset of parameters pertaining to the control curves (i.e., ε_R, H_R, γ, the low and high O/N scaling factors, and the two time offset parameters). These parameters were then constrained to their optimised values while the remaining model parameters were optimised to fit the Tum^+ MFL data. In final fits, both Tum^+ and Tum^- data were fitted and all parameters (including the control parameters) left unconstrained.

The final optimised parameters produced a good match to the data at almost all equilibration times as can be seen in Figure C.1. This came at the cost of permitting the the degradation rate of Tum to be slower than the dilution rate, as can be seen from the optimised parameters listed in the 'Full fit' column of Table C.1. Only by so doing could the deterministic model match the slow evolution of the high fluorescence sub-populations.

In this knowledge, an additional optimisation step was performed for just the long equilibration time data sets. This improved the discrepancy as can be seen from the 'Long only' column of Table C.1, but the change was still insufficient to allow the Tum half life to fall equal to or below the doubling time. As such, the 'Simple fit' strategy described in the main text (Section 6.3.1) was employed.

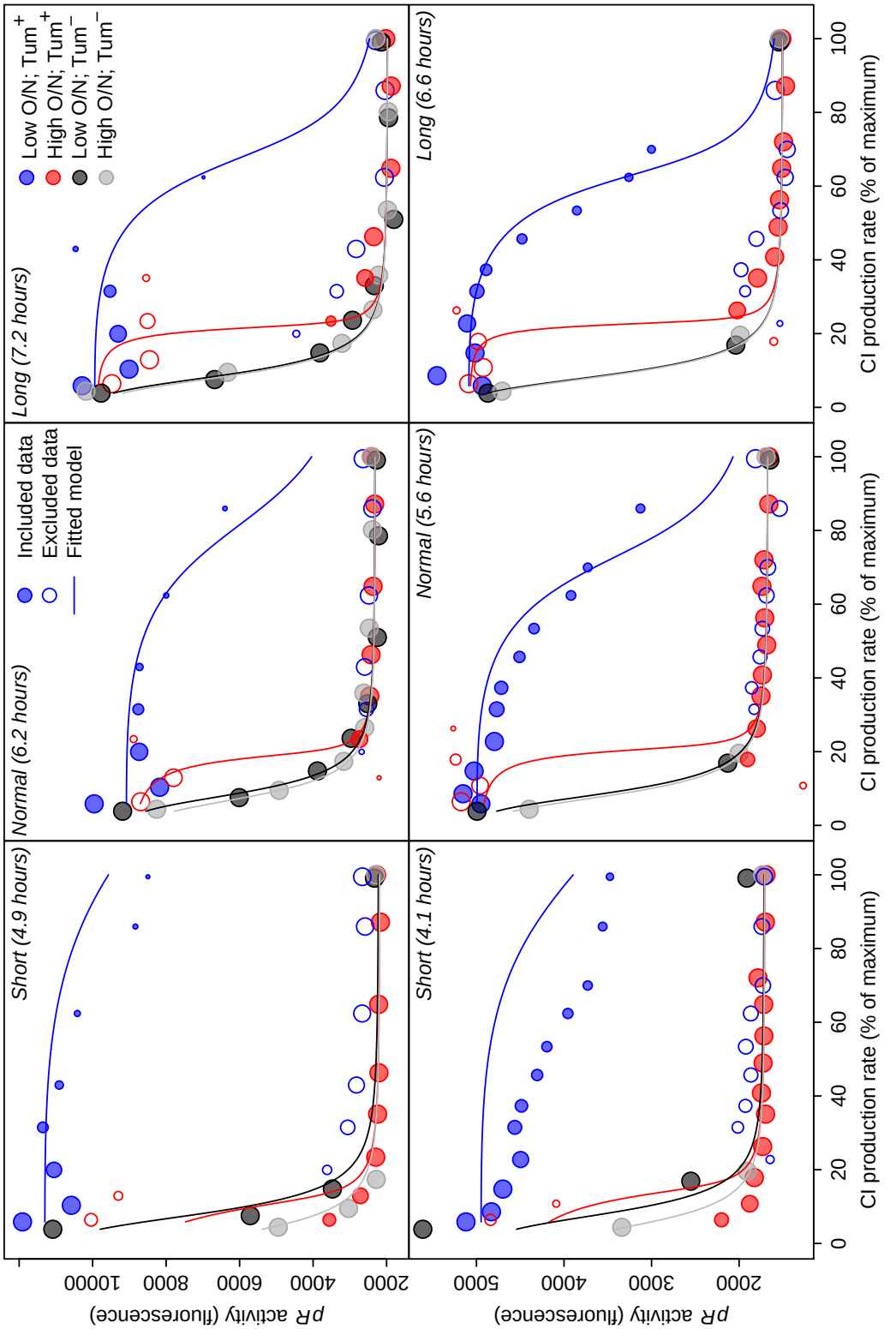


Figure C.1: Deterministic fits of the ER-MCTum–CI MFL flow cytometry hysteresis assay. *Continued on following page.*

Figure C.1 (*Continued*): Two sets of short, normal and long equilibration time hysteresis assays of the ER-MC strain were measured by flow cytometry: the first set (upper panels) were measured for lower cell counts (10,000–35,000), and the second set (lower panels) were measured for higher cell counts (78,000–92,000). The fluorescence distributions were curated as described in Figure 5.9 and the sub-populations are also plotted as described in that figure. Indicated in italic text for each assay is the equilibration type (short, normal or long) and the recorded number of hours of equilibration time. A deterministic model was fitted to the long equilibration time data as described in the text, but is shown overlaid in all panels for reference. Data points that were included in the parameter optimisation protocols are shown as filled circles, whilst those that were excluded are shown as open circles. As in previous figures, colour is used to distinguish history (low/high overnight (O/N)) and presence or absence of Tum (Tum^+/Tum^-). Since the scale of CI production was allowed to vary when fitting the model to the data, the units of the CI production rate are displayed as a percentage of the maximum CI production rate (i.e., the production rate of the high O/N Tum^+/Tum^- samples induced to 300 μM IPTG).

Table C.1: A comparison of the deterministic parameters fitted to various subsets of the time-course flow cytometry data. measured for the ER-MCTum−CI MFL. The upper block contains the core MFL parameters; the middle block contains the initial condition scaling parameters; the lower block contains the timing parameters. Unless otherwise indicated, limits specify 95% confidence intervals.

Parameter	**Full fit**[*]	**Long only**[**]	**Simple fit**[†]
ε_R (CI dimers/cell)	9.39×10^2	9.45×10^2	9.74×10^2
H_R	3.05	3.07	3.01
ε_S (Tum monomers/cell)	7.98×10^3	3.00×10^3	2.67×10^3
H_S	1.19	1.01	1.04
M_T (Tum monomers/cell)	3.19×10^4	3.19×10^4	3.18×10^4
Doubling time (min.)	27.1	27.0	32.4
Tum half life (min.)	153.5	51.5	32.4
Low O/N scaling	4.49	4.81	–
High O/N scaling	1.67	1.48	–
Initial Tum scaling	7.86	7.81	–
Assay time (hours)[‡]			
LCC, short equil.	2.4	2.4	–
LCC, normal equil.	3.7	3.7	–
LCC, long equil.	4.7	4.7	–
HCC, short equil.	3.0	2.8	–
HCC, normal equil.	4.5	4.3	–
HCC, long equil.	5.6	5.3	5.0

[*] Parameters determined experimentally in this thesis or derived from the literature.

[**] Parameters from the model shown in Figure 5.11 fitting all available parameters to match the time course data.

[†] Parameters from the model shown in Figure 6.3 fitted to match a single long equilibration time assay with a simplified parameter scheme.

[‡] The period of log phase growth attributed to each of the Low Cell Count (LCC) and High Cell Count (HCC) short, normal and long equilibration time (equil.) assays.

D

The hybrid stochastic/deterministic model of the Tum−CI MFL

Networks of stochastic reactions can be modelled as Markov processes with a probability density function that evolves according to a CME that defines the network [Gillespie, 1992; Kepler and Elston, 2001; Paulsson, 2005; Qian and Bishop, 2010]. In theory, the full stochastic behaviour of a gene network is completely specified by solutions to its CME, but deriving such solutions, whether analytically or numerically, is impractical for any but the simplest networks. Solutions must instead be characterised either via simulated trajectories consistent with the CME [Rao et al., 2002] or by making suitable approximations to the full CME [Thattai and van Oudenaarden, 2001; Elf and Ehrenberg, 2003].

In this thesis, stochastic modelling of the Tum−CI MFL is performed by running Monte-Carlo simulations of the network. Specifically, a hybrid stochastic/deterministic simulator [Kiehl et al., 2004] based on the Next-Reaction variant [Gibson and Bruck, 2000] of the standard Gillespie algorithm [Gillespie, 1977] is implemented. The basic theory and a summary of the modelled reactions has already been presented in Chapter 6. In this appendix, additional technical details and samples of the simulation code are included to explain how the simulator has been implemented.

The hybrid simulator extends the standard Gillespie algorithm by allowing for continuous changes to the state. To accommodate this in an otherwise discrete simulation method, the hazards (reaction propensities) dependent on any continuous processes need to be integrated over time to determine if changes in the deterministic variables affect the probability of getting the next reaction. Here, an Euler integration method is employed, which simplifies the implementation at the cost of reduced accuracy. However, by ensuring a small enough step size (set here to be at most 1 minute), the reduction in accuracy is inconsequential.

The core functionality of the stochastic model has been implemented in C++ and makes use of classes and inheritance to promote flexibility and reusability of the code. As such, the model has been split into various interchangeable units that are combined together to constitute the specific model. The most fundamental unit is the model state, which needs to synchronise between discrete and continuous representations of the system. A

state class was designed to handle these tasks in a transparent way and its code is listed in Appendix D.1.

The model itself is built from a 'state' class, a 'parameters' class, an 'equilibrator' class and a 'simulator' class. The syntax of these classes is specified in Appendix D.2 as a collection of interfaces. The 'simulator' class specifies the generic simulation method and comes in two forms: a basic form that takes equal time-steps and is suited to deterministic simulation, and a stochastic form that accepts a list of stochastic events that it can track and integrate the hazards of. The latter simulator is of most interest here, since it defines the hybrid simulation algorithm. The code implementing the Euler integration step of the hazards and coordinating event firing can be found at the end of the listing in Appendix D.2.

MFL-specific code can be found in Appendix D.3. This includes:

1. the definition of a class to store model parameters,
2. expanding the 'state' class to additionally track CI and/or Tum levels,
3. definitions of the stochastic events in the Tum−CI MFL, including cell division,
4. a highly optimised 'equilibrator' that solves the Tum−CI constraint equation (Equation (6.5)) and is called to determine the free species concentrations, and
5. declarations of the models (the deterministic tum^- control and MFL models, and the stochastic tum^- control and MFL models).

The code listings in Appendix D.4 exemplify how simulation is applied in practice. The example presented can be used to perform time-course simulations using the listed models and is part of an R library that allows the C++ library to be called from R.

The final section in this appendix (Appendix D.5) displays and explains the R code that was used to add experimental noise onto the raw stochastic simulation data. Mixture models of this generated data were fit using the constrained skew-*t* method described in Appendix B.

D.1 Tracking simulator state

As specified by Kiehl et al. [2004], it is important to ensure that discrete and continuous representations of the hybrid stochastic/deterministic model are kept well-synchronised. Here, the state of the system is defined to be completely characterised by the current volume V and three continuous state variables for the total concentrations of Tum, CI and GFP. The number of each protein is then determined by the product of that protein's concentration with the current volume, and discrete values are obtained by rounding down.

In the `stateinterfaces.h` header file, the 'Variable' class is defined to automatically detect whether a continuous or discrete representation is required based on the context. Operator overloading means that the 'Variable' class acts like a standard C++ variable.

The 'Variable' class depends on the 'Volume' class to convert between concentrations and counts, and changes in 'Volume' trigger all 'Variable' instances to update their recorded concentrations. The time, 'Volume' and 'Variable' instances are collected together in a 'state' structure to be passed between the various calculators of the simulator.

```
1  //////////////////////////////////////////////////////////////////////////
2  // stateinterfaces.h:
3  // Define interfaces for managing hybrid deterministic/stochastic simulation
4  // states. When defined this way, discrete and continuous representations of
5  // each state variable are maintained in sync.
6
7  #ifndef STATEINTERFACES_H
8  #define STATEINTERFACES_H
9
10 #include <cmath>
11 #include <cstdlib>
12 #include <vector>
13
14 typedef unsigned int uint;
15
16 class Volume;
17 class Variable;
18
19 // The relationship between discrete and continuous representations is
20 // dependent on cell volume. The Volume class keeps a list of registered
21 // simulation variables that should be notified when the volume is changed.
22 class Volume {
23   public:
24     Volume(double vol=1.0) : vol(vol) { }
25
26     // Variable registration and notification
27     void registerVariable(Variable&);
28     void volumeChanged();
29
30     // Type conversion to double simply accesses the volume
31     operator double() const { return vol; }
32
33     // Streamline definition of (compound) assignment operators using a macro:
34 #define VOLUME_ASSIGNMENT(Op)\
35     Volume& operator Op (const Volume& rhs) {\
36       vol Op rhs.vol; volumeChanged(); return *this;\
37     }\
38     Volume& operator Op (const double rhs) {\
39       vol Op rhs; volumeChanged(); return *this;\
40     }/* END VOLUME_ASSIGNMENT */
```

```
41
42      // Define the (compound) assignment operators
43      VOLUME_ASSIGNMENT(=)
44      VOLUME_ASSIGNMENT(+=)
45      VOLUME_ASSIGNMENT(-=)
46      VOLUME_ASSIGNMENT(*=)
47      VOLUME_ASSIGNMENT(/=)
48
49      // Streamline definition of arithmetic operators using a macro:
50  #define VOLUME_ARITHMETIC(Op,RetType)\
51      const RetType operator Op (const Volume& rarg) const {\
52        return vol Op rarg.vol;\
53      }\
54      friend const RetType operator Op (const Volume& larg, const Volume& rarg) {\
55        return larg.vol Op rarg.vol;\
56      }\
57      const RetType operator Op (const double rarg) const {\
58        return vol Op rarg;\
59      }\
60      friend const RetType operator Op (const double larg, const Volume& rarg) {\
61        return larg Op rarg.vol;\
62      }/* END VOLUME_ARITHMETIC */
63
64      // Define the arithmetic operators
65      VOLUME_ARITHMETIC(+,double)
66      VOLUME_ARITHMETIC(-,double)
67      VOLUME_ARITHMETIC(*,double)
68      VOLUME_ARITHMETIC(/,double)
69
70      // Define the relational operators
71      VOLUME_ARITHMETIC(<,bool)
72      VOLUME_ARITHMETIC(>,bool)
73      VOLUME_ARITHMETIC(<=,bool)
74      VOLUME_ARITHMETIC(>=,bool)
75      VOLUME_ARITHMETIC(==,bool)
76      VOLUME_ARITHMETIC(!=,bool)
77
78    // Printing
79    friend std::ostream& operator << (std::ostream& o, Volume& v) { o<<v.vol; return o; }
80
81    private:
82      double vol;
83      std::vector<Variable*> vars;
84      typedef std::vector<Variable*>::const_iterator varIter;
```

```
85 };
86
87 // Simulation variables that may be treated simultaneously in discrete and
88 // continuous spaces should be defined using the 'Variable' class. This class
89 // automates conversion between numbers of molecules and concentration. When
90 // operating using the 'double' type, the class works in concentrations; when
91 // operating using the 'unsigned int' type, the class works in molecule
92 // counts.
93 class Variable {
94   public:
95     // Initialise with a concentration:
96     Variable(double init, Volume& vol) : conc(init), vol(vol) {
97       fracCount = conc * vol;
98       // NB: casting from floating-point type to integer type is performed as
99       // truncation:
100      count = static_cast<uint>(std::abs(fracCount));
101      // Register to receive updates to the Volume:
102      vol.registerVariable(*this);
103    }
104    // Initialise with a count:
105    Variable(uint init, Volume& vol) : count(init), vol(vol) {
106      fracCount = static_cast<double>(count);
107      conc = fracCount / vol;
108      vol.registerVariable(*this);
109    }
110    // Initialise with an integer:
111    Variable(int init, Volume& vol) : vol(vol) {
112      // NB: negative values will be retained by fracCount
113      fracCount = static_cast<double>(init);
114      count = static_cast<uint>(std::abs(init));
115      conc = fracCount / vol;
116      vol.registerVariable(*this);
117    }
118
119    // Dual type conversions
120    operator uint() const { return count; }
121    operator int() const { return static_cast<int>(count); }
122    operator double() const { return conc; }
123
124    // Assignment
125    Variable& operator = (const Variable &rhs) {
126      count = rhs.count; fracCount = rhs.fracCount;
127      conc = rhs.conc; vol = rhs.vol; return *this;
128    }
```

```
129     Variable& operator = (const double rhs) {
130       conc = rhs; fracCount = rhs * vol;
131       count = static_cast<uint>(std::abs(fracCount));
132       return *this;
133     }
134     Variable& operator = (const uint rhs) {
135       count = rhs; fracCount = static_cast<double>(rhs);
136       conc = fracCount / vol; return *this;
137     }
138     Variable& operator = (const int rhs) {
139       fracCount = static_cast<double>(rhs);
140       count = static_cast<uint>(std::abs(rhs));
141       conc = fracCount / vol; return *this;
142     }
143
144     // Streamline definition of compound assignment operators using a macro:
145 #define VARIABLE_ASSIGNMENT(Op)\
146     Variable& operator Op (const Variable &rhs) {\
147       /* Operate on fracCounts, but derive other values using this */\
148       /* instance's volume: */\
149       fracCount Op rhs.fracCount;\
150       count = static_cast<uint>(std::abs(fracCount));\
151       conc = fracCount / vol; return *this;\
152     }\
153     Variable& operator Op (const double rhs) {\
154       conc Op rhs; fracCount = conc * vol;\
155       count = static_cast<uint>(std::abs(fracCount));\
156       return *this;\
157     }\
158     Variable& operator Op (const uint rhs) {\
159       count Op rhs; fracCount Op static_cast<double>(rhs);\
160       conc = fracCount / vol; return *this;\
161     }/* END VARIABLE_ASSIGNMENT */
162
163     // Define compound assignment operators
164     VARIABLE_ASSIGNMENT(+=)
165     VARIABLE_ASSIGNMENT(-=)
166     VARIABLE_ASSIGNMENT(*=)
167     VARIABLE_ASSIGNMENT(/=)
168
169     // Streamline definition of arithmetic operators using a macro:
170 #define VARIABLE_ARITHMETIC(Op, RetTypeConc, RetTypeCount)\
171     const RetTypeConc operator Op (const double rhs) const { return conc Op rhs; }\
172     friend const RetTypeConc operator Op (const double lhs, const Variable& rhs) {\
```

```
173       return lhs Op rhs.conc;\
174     }\
175     const RetTypeCount operator Op (const uint rhs) const { return count Op rhs; }\
176     friend const RetTypeCount operator Op (const uint lhs, const Variable& rhs) {\
177       return lhs Op rhs.count;\
178     }/* END VARIABLE_ARITHMETIC */
179
180     // Define arithmetic operators
181     VARIABLE_ARITHMETIC(+,double,uint)
182     VARIABLE_ARITHMETIC(-,double,uint)
183     VARIABLE_ARITHMETIC(*,double,uint)
184     VARIABLE_ARITHMETIC(/,double,uint)
185
186     // Define relational operators
187     VARIABLE_ARITHMETIC(>,bool,bool)
188     VARIABLE_ARITHMETIC(<,bool,bool)
189     VARIABLE_ARITHMETIC(>=,bool,bool)
190     VARIABLE_ARITHMETIC(<=,bool,bool)
191     VARIABLE_ARITHMETIC(==,bool,bool)
192     VARIABLE_ARITHMETIC(!=,bool,bool)
193
194     // Printing
195     friend std::ostream& operator << (std::ostream& o, Variable& v) { o<<v.conc; return o;
196
197     // This function gets called whenever the volume is updated:
198     void volumeChanged() {
199       // Only the concentration changes with volume changes:
200       conc = fracCount / vol;
201     }
202
203   private:
204     uint count;
205     double conc, fracCount;
206     Volume& vol;
207 };
208
209 // Given the definition of a Variable, now define Variable registration and
210 // event notification in the Volume class. Note that the Volume class is
211 // filled using references instantiated elsewhere, so is not responsible for
212 // creation or deletion of the registered Variables.
213 inline void Volume::registerVariable(Variable& var) {
214   vars.push_back(&var);
215 }
216 inline void Volume::volumeChanged() {
```

```
217     // Notify all registered variables that Volume has changed
218     for(varIter var = vars.begin(); var != vars.end(); ++var)
219       (*var)->volumeChanged();
220   }
221
222   // Simulation state is stored in the 'state' struct. Note that this struct can
223   // be copied and assigned to so that an initial state can be saved and then the
224   // main state later reset.
225   struct state {
226     explicit state() : t(0.0), vol(1.0), Report(0u, vol) { }
227     double t;
228     Volume vol;
229     Variable Report;
230     virtual double call_event(class stochEvent* event);
231   };
232
233   #endif /* STATEINTERFACES_H */
```

D.2 The generic simulation framework

```
1   ////////////////////////////////////////////////////////////////////////
2   // modelinterfaces.h:
3   // Define interfaces for accessing modelling simulators (stochastic or
4   // deterministic). Inherit from these interfaces to create modelling protocols
5   // with defined operation.
6
7   #ifndef MODELINTERFACES_H
8   #define MODELINTERFACES_H
9
10  #include <gsl/gsl_rng.h>
11  #include <vector>
12  #include "stateinterfaces.h"
13
14  // To avoid divisions by zero and numerical errors we need to set volatile
15  // concentrations close to zero to some absolute zero reference.
16  const double zeroconc = 1e−3;
17  // Use a different threshold for zero for hazards
18  const double zerohazard = 1e−9;
19
20  // Parameter lists should be noncopyable to avoid duplicate parameter sets
21  // diverging inappropriately during initialisation
22  struct parameters {
23    protected:
24      parameters() {}
```

```
25      ~parameters() {}
26
27    private:
28      parameters(const parameters &);
29      parameters & operator = (const parameters &);
30  };
31
32  // The equilibrator class needs an equilibrate and steadystate function
33  // and typically should need no more than a pointer to some state and a
34  // pointer to some parameter set from which to make calculations
35  class equilibrator {
36    public:
37      virtual void equilibrate() = 0;
38      virtual void steadystate() = 0;
39
40    protected:
41      // Interface definition, so keep the constructor protected
42      explicit equilibrator(state* s,parameters* p) : s(s),p(p) { }
43      // We do not want to allow polymorphic deletion
44      virtual ~equilibrator() { }
45
46    private:
47      // noncopyable
48      equilibrator(const equilibrator &);
49      equilibrator & operator = (const equilibrator &);
50
51      // Keep these private so that derived classes can alternatively work with
52      // children of 'state' or 'parameters' classes
53      state* s;
54      parameters* p;
55  };
56
57  ////////////////////////////////////////////////////////////////////////
58  // 'simulator' class
59  // ****************
60  // The base simulation class from which new simulation classes should be
61  // derived. You can either request single steps to be taken, or specify a time
62  // to simulate to (the class then works out how many steps it needs to take).
63
64  class simulator {
65    public:
66      // Simulation
67      virtual void initialise() { } // Optional initialisation function
68      virtual void step(double ts) { s->t += ts; eq->equilibrate(); }
```

```
69      virtual void simulate(double simtime) {
70        // NB: we want to simulate an ADDITIONAL simtime minutes
71        simtime += s->t;
72
73        // Step the default time step size unless it would take us past simtime:
74        while( (simtime - s->t) > ts_default ) step(ts_default);
75
76        // Finish by stepping exactly to the desired step time. This is in a
77        // while loop since the timestep supplied to step() is preferred but
78        // not guaranteed:
79        while(s->t < simtime) step(simtime - s->t);
80      }
81
82      // We do not want to allow polymorphic deletion but to avoid warnings make
83      // destructor virtual and public...
84      virtual ~simulator() { }
85
86    protected:
87      // Interface definition, so keep the constructor protected
88      simulator(state* s, parameters* p, equilibrator* eq, const double ts)
89        : ts_default(ts), s(s), p(p), eq(eq) { }
90
91      // The default time step size:
92      const double ts_default;
93
94    private:
95      // noncopyable
96      simulator(const simulator &);
97      simulator & operator = (const simulator &);
98
99      // Keep these private so that derived classes can alternatively work with
100     // children of 'state', 'parameters' or 'equilibrator' classes
101     state* s;
102     parameters* p;
103     equilibrator* eq;
104 };
105
106 ////////////////////////////////////////////////////////////////////////
107 // 'model' class
108 // *************
109 // This base class specifies how all the components should be collected
110 // together for a complete modelling package. This is the interface that users
111 // will access. A model should be constructed with a private collection of the
112 // unique instances of 'state', 'parameters', 'equilibrator' and 'simulator'
```

```
113 // classes
114
115 class model {
116   public:
117     // 'initialise' member functions should do something equivalent to the
118     // following but may need updating depending on the state struct:
119     virtual void initialise() {
120       s->t = 0.0; s->Report = 0.0; // Set all state variables
121       eq->steadystate(); // Relax initial values to steadystate
122       sim->initialise(); // Run any simulator initialisation
123     }
124
125     // Accessors & mutators
126     virtual double time() { return s->t; }
127     virtual void time(double t) { s->t = t; }
128     virtual double activity() { return s->Report; }
129
130     // Equilibration
131     virtual void equilibrate() { eq->equilibrate(); }
132     virtual void steadystate() { eq->steadystate(); }
133
134     // Simulation
135     virtual void step(double ts) { sim->step(ts); }
136     virtual void simulate(double time) { sim->simulate(time); }
137
138     // We do not want to allow polymorphic deletion but to avoid warnings make
139     // destructor virtual and public...
140     virtual ~model() { }
141
142   protected:
143     // Protect the constructor so that only children can initialise a model
144     explicit model(state* s,parameters* p,equilibrator* eq,simulator* sim)
145       : s(s), p(p), eq(eq), sim(sim) { }
146
147   private:
148     // noncopyable
149     model(const model &);
150     model & operator = (const model &);
151
152     // Keep these private so that derived classes can alternatively work with
153     // children of 'state', 'parameters', 'equilibrator' or 'simulator'
154     // classes
155     state* s;
156     parameters* p;
```

```
157     equilibrator* eq;
158     simulator* sim;
159 };
160
161 //////////////////////////////////////////////////////////////////////////
162 // 'stochEvent' class
163 // ******************
164 // Inherit from this class to build new stochastic events. An event needs the
165 // following members:
166 // − a function that samples this event's probability distribution
167 // − a comparison operator so that we can choose the next event
168 // − a function to update the state
169 // − a pointer to a RNG
170 // − Note that step() should NOT update the state time
171
172 class stochEvent {
173   friend bool compare(stochEvent*, stochEvent*);
174
175   public:
176
177   /*** Event−specific behaviour − children must define these functions ***/
178
179   // Override these to return the instantaneous hazard for this event
180   virtual double hazard(state* init) {
181     throw "Error: base stochEvent::hazard(state*)  method called.";
182   }
183   virtual double hazard(struct repress* init) {
184     // Delegate to children by upcasting
185     return this->hazard((state*)init);
186     /* throw "Error: base stochEvent::hazard(repress*) method called."; */
187   }
188   virtual double hazard(struct mfl* init) {
189     // Delegate to children by upcasting
190     return this->hazard((repress*)init);
191     /* throw "Error: base stochEvent::hazard(mfl*) method called."; */
192   }
193
194   // Fire the event, updating the global state
195   virtual void fire() = 0;
196
197   /*** Standard stochastic event behaviour ***/
198
199   // Event behaviour is assumed Poissonian, so sample a random number from
200   // Exp(1) and reset the cumulative hazard
```

```
201   virtual void sample() {
202     rexp = -log(gsl_rng_uniform_pos(r));
203     h_cumulative = 0;
204   }
205
206   // Peek ahead to see if this event will fire within the supplied time step
207   virtual bool will_fire(double ts) {
208     if(h_cumulative + ts * hazard(s) > rexp)
209       return true;
210     else
211       return false;
212   }
213
214   // Return the time to the next event assuming a Euler timestep
215   virtual double time_to_event() {
216     double h = hazard(s);
217     // If hazard is too close to zero (or negative), set it to a very small
218     // (nonzero) value to guarantee the event will be outside the current time
219     // step:
220     if(h < zerohazard) h = zerohazard;
221     /* std::cout<<"Event: "<<this->name()<<"; hazard: "<<h; */
222     /* std::cout<<"; cumulative hazard: "<<h_cumulative; */
223     /* std::cout<<"; random draw: "<<rexp<<std::endl; */
224     return (rexp - h_cumulative)/h;
225   }
226
227   // Step the cumulative hazard forward in time
228   virtual void step(double ts, state* init) {
229     // Update the cumulative hazard:
230     h_cumulative += ts * hazard(init);
231   }
232
233   virtual const char* name() { return "unspecified"; }
234
235   // We do want to allow polymorphic deletion so we need a virtual
236   // destructor to ensure that deletion is defined
237   virtual ~stochEvent() { }
238
239   protected:
240   // Interface definition, so keep the constructor protected
241   explicit stochEvent(state* s, parameters* p, gsl_rng* r)
242     : rexp(0.0), h_cumulative(0.0), r(r), s(s), p(p) { }
243
244   double rexp; // Random draw from Exp(1) distribution
```

```
245   double h_cumulative; // Cumulative hazard
246   gsl_rng* r;
247
248   private:
249   // noncopyable
250   stochEvent(const stochEvent &);
251   stochEvent & operator = (const stochEvent &);
252
253   // Keep these private so that derived classes can alternatively work with
254   // children of 'state' or 'parameters' classes
255   state* s;
256   parameters* p;
257 };
258
259 // Define a compare function of pointers to 'stochEvent' objects since we will
260 // want to find the next event according to its 't_event' from a vector of
261 // pointers to stochEvent objects
262 inline bool compare(stochEvent* a, stochEvent* b)
263   { return a->time_to_event() < b->time_to_event(); }
264
265 // Allow double dispatch between events and states:
266 inline double state::call_event(stochEvent* event) { return event->hazard(this); }
267
268 //////////////////////////////////////////////////////////////////////////
269 // 'stochSimulator' base class
270 // ***************************
271 // Stochastic simulators should inherit from this class by allocating a set of
272 // possible stochastic events to the events vector.
273
274 template<class StateType> class stochSimulator : public simulator {
275   public:
276     virtual void initialise();
277     virtual inline void step(double);
278     // Add in an optional deterministic step
279     virtual inline void step_detrm(double ts) { };
280
281   protected:
282     // Interface definition, so keep the constructor and destructor protected
283     explicit stochSimulator(StateType* s, parameters* p,
284         equilibrator* eq, gsl_rng* r, double ts=1.0)
285       : simulator(s,p,eq,ts), r(r), s(s), eq(eq) { }
286     ~stochSimulator() {
287       // Ensure that allocated events are destroyed:
288       if(!events.empty())
```

```
289   for(eventIterator iter = events.begin(); iter != events.end(); ++iter)
290     delete (*iter);
291     }
292
293     gsl_rng* r;
294     std::vector<stochEvent*> events; // Vector of possible stochastic events
295     typedef std::vector<stochEvent*>::const_iterator eventIterator;
296
297   private:
298     // noncopyable by inheritance
299
300     StateType* s;
301     equilibrator* eq;
302 };
303
304 template<class StateType>
305 inline void stochSimulator<StateType>::initialise() {
306   // Make sure we don't initialise (or start) simulation if our simulator has
307   // no events (otherwise we will get segmentation faults...)
308   if(events.empty())
309     throw "Error in stochastic simulator: events stack is empty!";
310
311   // Draw random times from each of the distributions
312   std::for_each(events.begin(),events.end(),std::mem_fun(&stochEvent::sample));
313 }
314
315 template<class StateType>
316 inline void stochSimulator<StateType>::step(double ts){
317   // This step is essentially an ODE integration step
318   // For the moment, Euler integration steps are taken to keep the algorithm
319   // simple (later it might be good to perform Runge-Kutta integration).
320   // The simulator needs to update the deterministic reactions, but also
321   // needs to integrate over any time-varying stochastic hazards.
322
323   // We need to save the initial state so that it can be re-used to update
324   // both the deterministic reactions and evolution of the hazards:
325   StateType init(*s);
326
327   // First check if any events are due for firing:
328   eventIterator will_fire;
329   for(will_fire = events.begin(); will_fire != events.end(); ++will_fire)
330     if( (*will_fire)->will_fire(ts) ) break;
331
332   if(will_fire != events.end()){
```

```
333       // An event will fire this step, so estimate event firing times and pick
334       // the minimum:
335       eventIterator nextEvent = std::min_element(events.begin(),events.end(),&compare);
336
337       double ts_event = (*nextEvent)->time_to_event();
338
339       // First step the deterministic reactions forwards to the event:
340       step_detrm(ts_event);
341
342       // Then step each stochastic hazard forwards to the event:
343       for(eventIterator iter = events.begin(); iter != events.end(); ++iter)
344         (*iter)->step(ts_event, &init);
345
346       // Update the state time:
347       s->t += ts_event;
348
349       // Finally, take the stochastic jump:
350       (*nextEvent)->fire();
351       // And sample a new random number:
352       (*nextEvent)->sample();
353     } else {
354       // First step the deterministic reactions forwards:
355       step_detrm(ts);
356
357       // Then step each stochastic hazard forwards:
358       for(eventIterator iter = events.begin(); iter != events.end(); ++iter)
359         (*iter)->step(ts, &init);
360       // Update the state time
361       s->t += ts;
362     }
363
364     // Once the ODE integration step has completed, finish by equilibrating
365     // any fast reactions:
366     eq->equilibrate();
367 }
368
369 #endif /* MODELINTERFACES_H */
```

D.3 Classes for simulation of the MFL

D.3.1 mflLibrary.h

The first part of `mflLibrary.h` defines C++ structures that contain parameters used when modelling the Tum−CI MFL. A number of different parameter sets were used, but the first exemplifies the general syntax of these structures. The other parameter sets are omitted

for brevity.

```
1  //////////////////////////////////////////////////////////////////////////
2  // mflLibrary.h:
3  // Define the procedures for modelling MFL-like genetic networks.
4
5  #ifndef MFLLIBRARY_H
6  #define MFLLIBRARY_H
7
8  #include "modelinterfaces.h"
9  #include <cmath>
10 #include <gsl/gsl_rng.h>
11 #include <vector>
12
13 //////////////////////////////////////////////////////////////////////////
14 // 'MFLparams' structure
15 // *********************
16 // A structure for holding all the constants necessary for making calculations
17 // from a given state.
18
19 struct MFLparams : public parameters {
20   // Copy numbers
21   double genome, PtCopy, pRcopy, cSites, nM_per_atom, ConcCIsites;
22   // Parameters used for modelling cell volume:
23   double Tdbl, Vinit, Vrate, PartProb, Vscale, VrateDt, PartProbDt;
24   // Production and degradation
25   double PcBasal, PcMax, Pt, Pz, Dc, Dt, Dz, Mz, units_per_conc;
26   // Parameters used when fitting the model to data:
27   double PtInitScale, DtScale, LOscale, HOscale;
28   // Stochastic parameters
29   double PcMaxBurstRate, PtMaxBurstRate, PzMaxBurstRate;
30   double PcBurstSize, PtBurstSize, PzBurstSize;
31   double DegEvents;
32   // Equilibria Parameters
33   double Hr, Er, Hs, Es;
34   // NB: the following parameters are not currently used.
35   double Ds, kson, ksoff;
36
37   void print(std::ostream& out = std::cout) const {
38     out<<"**** Copy Numbers ****"<<std::endl;
39     out<<"Genome: "<<genome<<"; PtCopy: "<<PtCopy<<"; pRcopy: "<<pRcopy;
40     out<<"; cSites: "<<cSites<<"; ConcCIsites: "<<ConcCIsites<<std::endl;
41
42     out<<"**** Modelling cell volume ****"<<std::endl;
43     out<<"Doubling time: "<<Tdbl<<"; Vinit: "<<Vinit<<"; Vscale: "<<Vscale;
```

```
44      out<<"; Vrate: "<<Vrate<<"; VrateDt: "<<VrateDt<<std::endl;
45      out<<"; PartProb: "<<PartProb<<"; PartProbDt: "<<PartProbDt<<std::endl;
46
47      out<<"**** Production (nM/min) ****"<<std::endl;
48      out<<"PcBasal: "<<PcBasal<<"; PcMax: "<<PcMax<<"; Pt: "<<Pt<<std::endl;
49      out<<"**** LacZ Production (Units/min) ****"<<std::endl;
50      out<<"units_per_conc: "<<units_per_conc<<"; Pz: "<<Pz<<std::endl;
51
52      out<<"**** Parameters used to fit the model to data ****"<<std::endl;
53      out<<"PtInitScale: "<<PtInitScale<<"; DtScale: "<<DtScale;
54      out<<"; LOscale: "<<LOscale<<"; HOscale: "<<HOscale<<std::endl;
55
56      out<<"**** Degradation ****"<<std::endl;
57      out<<"Dc: "<<Dc<<"; Dt: "<<Dt<<"; Ds: "<<Ds<<"; Dz: "<<Dz<<std::endl;
58
59      out<<"**** Stochastic ****"<<std::endl;
60      out<<"PcMaxBurstRate: "<<PcMaxBurstRate<<"; PcBurstSize: "<<PcBurstSize<<std::endl;
61      out<<"PtMaxBurstRate: "<<PtMaxBurstRate<<"; PtBurstSize: "<<PtBurstSize<<std::endl;
62      out<<"PzMaxBurstRate: "<<PzMaxBurstRate<<"; PzBurstSize: "<<PzBurstSize<<std::endl;
63      out<<"DegEvents: "<<DegEvents<<" per minute; relative degSizes: ";
64      out<<100*Dc/DegEvents<<"% (Ct), "<<100*Dt/DegEvents<<"% (Tt)"<<std::endl;
65      out<<"Ct degSizes in molecules: "<<PcBasal/(DegEvents*nM_per_atom)<<" (min), ";
66      out<<PcMax/(DegEvents*nM_per_atom)<<" (max)"<<std::endl;
67      out<<"Tt degSizes in molecules: "<<Pt/((1 + std::pow(PcMax/(Dc*Er),Hr))*DegEvents*nM_per_atom
68      out<<" (min), "<<Pt/(DegEvents*nM_per_atom)<<" (max)"<<std::endl;
69
70      out<<"**** Equilibria ****"<<std::endl;
71      out<<"Hr: "<<Hr<<"; Er: "<<Er<<"; Hs: "<<Hs<<"; Es: "<<Es<<std::endl;
72      out<<"kson: "<<kson<<"; ksoff: "<<ksoff<<std::endl;
73    }
74
75    explicit MFLparams(bool TumDeg = true) {
76      defaultInit();
77      if(!TumDeg) changeDt(Dc);
78    }
79
80    explicit MFLparams(double PcBS,double PtBS,double NewEs=1175.0,double newDt=0.0693) {
81      defaultInit();
82      changeDt(newDt);
83      Es = NewEs;
84
85      // Estimated burst sizes
86      PcBurstSize = PcBS;
87      PtBurstSize = PtBS;
```

```
88      // Calculated burst rates
89      PcMaxBurstRate = PcMax/PcBurstSize;
90      PtMaxBurstRate = Pt/PtBurstSize;
91    }
92
93    explicit MFLparams(double newDt) {
94      defaultInit();
95      changeDt(newDt);
96    }
97
98
99    protected:
100   void changeDt(double newDt) {
101     Pt = Pt/Dt*newDt;
102     Dt = newDt;
103     PtBurstSize = Pt/PtMaxBurstRate; // (nM)
104     units_per_conc = Pz/Pt;
105   }
106
107   private:
108   void defaultInit() {
109     // DNA elements
110     genome =  2.3; // Number of copies of the genome per cell
111     PtCopy = 10.0; // Number of copies of pR-tum relative to the genome
112     pRcopy = 12.0; // Number of copies of pR relative to the genome
113     cSites =  7.0; // Number of CI dimer binding sites per pR promoter
114     nM_per_atom = 1.47; // Conversion from absolute number per cell to nM
115
116     // Calculate the concentration of available sites for CI dimer binding
117     ConcCIsites = genome * pRcopy * cSites * nM_per_atom;
118
119     /***** Production & Degradation *****/
120
121     // Production rates (nM/min)
122     PcBasal = 14.6; PcMax = 326.8; Pt =  2050.0;
123
124     // Degradation rates (min^-1)
125     Dc = 0.0173; Dt = 0.0693;
126
127     // Degradation rate of lacZ is the same as CI
128     Dz = Dc;
129     // Production rate of lacZ is the unrepressed steadystate value (in units)
130     // multiplied by the degradation rate.
131     Pz = 850.0 * Dz;
```

```
132     // The lacZ transcription rate (in units/min) can be determined from the
133     // Pt promoter production rate (in nM/min, where Pt gives the maximum
134     // rate) through a conversion via units_per_conc.
135     units_per_conc = Pz / Pt;
136
137     /***** Stochastic Parameters *****/
138
139     // Estimated burst rates in RNAP firings per minute
140     PcMaxBurstRate = 4.3;
141     PzMaxBurstRate = 7.3; // NB: value for single copy pR reporter
142     PtMaxBurstRate = PtCopy*PzMaxBurstRate; // NB: value for PtCopies of pR
143
144     // Calculated burst sizes
145     PcBurstSize = PcMax/PcMaxBurstRate; // (nM)
146     PtBurstSize = Pt/PtMaxBurstRate; // (nM)
147     PzBurstSize = Pz/PzMaxBurstRate; // (units)
148
149     // Number of stochastic degradation events to occur per minute
150     DegEvents = 10.0; // per minute
151     // NB: to ensure different timescales for degradation and production, set
152     // this parameter to 600 to get 10 degradation events per second. 20 per
153     // minute will mean that at low firing rates, continuous degradation will
154     // never occur over a timescale where the consequent change in PtRate will
155     // have an impact.
156
157     /***** Equilibria Parameters *****/
158
159     // pR repression
160     Hr = 1.2; Er =  690.0; // (nM)
161     // Tum−CI binding
162     Hs = 2.5; Es = 1175.0; // (nM)
163
164     /***** Unused Parameters *****/
165
166     Ds =     0.0173; // (min^−1)
167     kson =     3.73; // (nM^−Hs.min^−1)
168     ksoff = 1.77e+8; // (min^−1)
169   }
170 };
```

After defining the parameter sets, `mflLibrary.h` continues by defining state structures specific to the MFL, and a cohort of stochastic events that can be used in various formulations of the stochastic model.

```
488 //////////////////////////////////////////////////////////////////////////
```

```
489 // State structures
490 // ***************
491 // Structures for containing the state of the model
492
493 struct repress : virtual public state {
494   explicit repress()
495     : PcR(10.0), PtR(10.0), Ct(10u, vol) { }
496   double PcR;
497   double PtR;
498   Variable Ct;
499   // Allow double dispatch between events and states:
500   double call_event(stochEvent* event) { return event->hazard(this); }
501 };
502
503 struct mfl : virtual public repress {
504   explicit mfl() : Tt(10u, vol) { }
505   Variable Tt;
506   // Allow double dispatch between events and states:
507   double call_event(stochEvent* event) { return event->hazard(this); }
508 };
509
510 /*****************************************************************************
511   Stochastic Event Definitions
512 *****************************************************************************/
513
514 /***** Pc Production variants *****/
515
516 // Stochastic production from Pc. Burst size constant.
517 class stochPcNoMsg : public stochEvent {
518   public:
519     stochPcNoMsg(repress* s, MFLparams* p, gsl_rng* r)
520       : stochEvent(s,p,r), p(p), s(s)
521     { h_scale = 1/p->PcBurstSize; }
522
523     // Double dispatch to the state-specific function
524     double hazard(state* init) { return init->call_event(this); }
525     double hazard(repress* init) { return init->PcR * h_scale; }
526     double hazard(mfl* init) {
527       /* std::cout<<"stochPcNoMsg- PcR: "<<init->PcR; */
528       /* std::cout<<"; h_scale: "<<h_scale<<std::endl; */
529       return init->PcR * h_scale; }
530
531     void fire(){
532       s->Ct += p->PcBurstSize;
```

```
533     }
534
535   protected: MFLparams* p; double h_scale;
536   private: repress* s;
537 };
538
539 // Stochastic production from Pc. Burst size stochastic.
540 class stochPc : public stochPcNoMsg {
541   public:
542     stochPc(repress* s, MFLparams* p, gsl_rng* r)
543       : stochPcNoMsg(s,p,r), s(s) { }
544     void fire() {
545       s->Ct += p->PcBurstSize * (-log(gsl_rng_uniform_pos(r)));
546     }
547     const char* name() { return "stochPc"; }
548   private: repress* s;
549 };
550
551 // Truly discrete production of CI molecules (i.e., molecules are added
552 // as counts and not as volume-dependent concentrations).
553 class stochPcDiscrete : public stochPc {
554   public:
555     stochPcDiscrete(repress* s, MFLparams* p, gsl_rng* r)
556       : stochPc(s,p,r), s(s) {
557         p_geom = p->Vscale/p->PcBurstSize;
558       }
559     void fire() {
560       // For truly discrete production, molecules should be added as counts
561       // (i.e., not added as concentrations that have a volume dependence). So
562       // choose a burst size from the geometric distribution with mean
563       // 1/p_geom = 1.5*Vinit*PcBurstSize (1.5*Vinit is the mean cell volume
564       // each generation, so should correctly scale the PcBurstSize from
565       // molecules/cell to a count):
566       s->Ct += gsl_ran_geometric(r, p_geom);
567     }
568   protected: double p_geom;
569   private: repress* s;
570 };
571
572 /***** Pt Production variants *****/
573
574 // Stochastic reporter production from Pt. Burst size stochastic.
575 class stochPtReport : public stochEvent {
576   public:
```

```
577     stochPtReport(repress* s, MFLparams* p, gsl_rng* r)
578       : stochEvent(s,p,r), p(p), s(s)
579     { h_conversion = p->units_per_conc/p->PzBurstSize; }
580
581     // Double dispatch to the state-specific function
582     double hazard(state* init) { return init->call_event(this); }
583     // Instantaneous hazard of reporter production at a given state:
584     double hazard(repress* init) {
585       // The Pt rate (monomers) must be converted to reporter units
586       return init->PtR * h_conversion;
587     }
588     double hazard(mfl* init) {
589       // The Pt rate (monomers) must be converted to reporter units
590       return init->PtR * h_conversion;
591       /* std::cout<<"stochPtReport; conversion: "<<h_conversion; */
592       /* std::cout<<"PtR: "<<init->PtR<<std::endl; */
593     }
594
595     void fire() {
596       s->Report += p->PzBurstSize * (-log(gsl_rng_uniform_pos(r)));
597     }
598
599     const char* name() { return "stochPtReport"; }
600   protected: double h_conversion; MFLparams* p;
601   private: repress* s;
602 };
603
604 // Truly discrete production of reporter molecules (i.e., molecules are added
605 // as counts and not as volume-dependent concentrations).
606 class stochPtReportDiscrete : public stochPtReport {
607   public:
608     stochPtReportDiscrete(repress* s, MFLparams* p, gsl_rng* r)
609       : stochPtReport(s,p,r), s(s) {
610         p_geom = p->Vscale/p->PzBurstSize;
611       }
612     void fire() {
613       // For truly discrete production, molecules should be added as counts
614       // (i.e., not added as concentrations that have a volume dependence). So
615       // choose a burst size from the geometric distribution with mean
616       // 1/p_geom = 1.5*Vinit*PzBurstSize (1.5*Vinit is the mean cell volume
617       // each generation, so should correctly scale the PzBurstSize from
618       // molecules/cell to a count):
619       s->Report += gsl_ran_geometric(r, p_geom);
620     }
```

```
621   protected: double p_geom;
622   private: repress* s;
623 };
624
625 // Stochastic Tum production from Pt. Burst size stochastic.
626 class stochPtTum : public stochEvent {
627   public:
628     stochPtTum(mfl* s, MFLparams* p, gsl_rng* r)
629       : stochEvent(s,p,r), p(p), s(s)
630     { h_scale = 1/p->PtBurstSize; }
631
632     // Double dispatch to the state-specific function
633     double hazard(state* init) { return init->call_event(this); }
634     double hazard(mfl* init) {
635       return init->PtR * h_scale;
636     }
637
638     void fire() {
639       s->Tt += p->PtBurstSize * (-log(gsl_rng_uniform_pos(r)));
640     }
641
642     const char* name() { return "stochPtTum"; }
643
644   protected: double h_scale; MFLparams* p;
645   private: mfl* s;
646 };
647
648 // Truly discrete production of Tum molecules (i.e., molecules are added as
649 // counts and not as volume-dependent concentrations).
650 class stochPtTumDiscrete : public stochPtTum {
651   public:
652     stochPtTumDiscrete(mfl* s, MFLparams* p, gsl_rng* r)
653       : stochPtTum(s,p,r), s(s) {
654         p_geom = p->Vscale/p->PtBurstSize;
655       }
656     void fire() {
657       // For truly discrete production, molecules should be added as counts
658       // (i.e., not added as concentrations that have a volume dependence). So
659       // choose a burst size from the geometric distribution with mean
660       // 1/p_geom = 1.5*Vinit*PtBurstSize (1.5*Vinit is the mean cell volume
661       // each generation, so should correctly scale the PtBurstSize from
662       // molecules/cell to a count):
663       s->Tt += gsl_ran_geometric(r, p_geom);
664     }
```

```
665   protected: double p_geom;
666   private: mfl* s;
667 };
668
669 /***** Cell division *****/
670
671 // Here, cell division is defined to be a non-stochastic event that occurs
672 // regularly after each period of cell division. So override each of the
673 // stochastic functions, replacing with deterministic ones:
674 class cellDivision : public stochEvent {
675   public:
676     cellDivision(state* s, MFLparams* p, gsl_rng* r)
677       : stochEvent(s,p,r), s(s) {
678         // The hazard here is simply the rate of growth in volume:
679         h_scale = p->Vrate;
680         // The initial volume is also the cap on h_cumulative, since cells
681         // double in size:
682         Vinit = p->Vinit;
683         // The probability used for partitioning species by the binomial
684         // distribution:
685         PartProb = p->PartProb;
686       }
687
688     // Override the sampling function to make cell division deterministically
689     // periodic:
690     void sample() {
691       h_cumulative = 0; // Reset the volume growth counter
692       rexp = Vinit; // 'rexp' in this instance is the target growth in volume
693     }
694
695     // Hazard has no state dependence, so do not need to double dispatch in
696     // this case:
697     double hazard(state* init) { return h_scale; }
698     double hazard(repress* init) { return h_scale; }
699     double hazard(mfl* init) { return h_scale; }
700
701     void fire() {
702       // Reset the volume to initial:
703       s->vol = Vinit;
704
705       // Split each population in half according to the binomial distribution:
706       s->Report = gsl_ran_binomial(r, PartProb, s->Report);
707       // NB: the Variable class automatically handles conversion to and from
708       // unsigned ints in the above.
```

```
709       }
710     protected: double h_scale, PartProb, Vinit, Vfinal;
711     private: state* s;
712   };
713
714   class cellDivisionRepress : public cellDivision {
715     public:
716       cellDivisionRepress(repress* s, MFLparams* p, gsl_rng* r)
717         : cellDivision(s,p,r), s(s) { }
718       void fire() {
719         /* std::cout<<"** Before split **  Vol: "<<s->vol; */
720         /* std::cout<<"; Ct: "<<s->Ct<<" / "<<static_cast<uint>(s->Ct)<<std::endl; */
721
722         cellDivision::fire(); // Run the parent event first
723         // Additionally split CI population in half:
724         s->Ct = gsl_ran_binomial(r, PartProb, s->Ct);
725
726         /* std::cout<<"** After split  **  Vol: "<<s->vol; */
727         /* std::cout<<"; Ct: "<<s->Ct<<" / "<<static_cast<uint>(s->Ct)<<std::endl; */
728       }
729     private: repress* s;
730   };
731
732   class cellDivisionMFL : public cellDivisionRepress {
733     public:
734       cellDivisionMFL(mfl* s, MFLparams* p, gsl_rng* r)
735         : cellDivisionRepress(s,p,r), s(s) { }
736       void fire() {
737         cellDivisionRepress::fire(); // Run the parent event first
738         // Additionally split Tum population in half:
739         s->Tt = gsl_ran_binomial(r, PartProb, s->Tt);
740       }
741     private: mfl* s;
742   };
743
744   #endif /* MFLLIBRARY_H */
```

D.3.2 equilibration.h

```
1  //////////////////////////////////////////////////////////////////////
2  // equilibration.h:
3  // Define protocols for equilibrating the Tum-CI and CI-pR equilibria in the
4  // MFL strains.
5
```

```
6  #ifndef EQUILIBRATION_H
7  #define EQUILIBRATION_H
8
9  #include "mflLibrary.h"
10 #include <gsl/gsl_roots.h>
11 #include <vector>
12 #include <algorithm>
13
14 // To save on computing time, set a threshold squared difference that is
15 // tolerable when recalculating the hill coefficient terms.
16 const double diffthreshold = 1e-12;
17
18 /*************************************************************************/
19 // 'gslRootSolver' class
20 // ********************
21 // Class to wrap up the root solving algorithm from the GSL
22
23 struct gslError {
24   gslError(int error_code,const char* function)
25     : error_code(error_code),function(function) { }
26   int error_code;
27   const char* function;
28 };
29
30 class gslRootSolver {
31   public:
32     explicit gslRootSolver(double rel_tol = 0.0001,unsigned int max_iter = 100)
33       : rel_tol(rel_tol),max_iter(max_iter),solverType(gsl_root_fsolver_brent)
34       { solver = gsl_root_fsolver_alloc(solverType); }
35     ~gslRootSolver()
36       { gsl_root_fsolver_free(solver); }
37
38     double tolerance() { return rel_tol; }
39     unsigned int maxIterations() { return max_iter; }
40     unsigned int prevIterations() { return iter; }
41
42     double findroot(gsl_function*,double,double);
43
44   private:
45     double rel_tol;
46     unsigned int iter;
47     unsigned int max_iter;
48     const gsl_root_fsolver_type* solverType;
49     gsl_root_fsolver *solver;
```

```
50 };
51
52 inline double gslRootSolver::findroot(gsl_function *F,double initLower,double initUpper) {
53   // std::cout<<initLower<<" < "<<initUpper<<std::endl;
54   iter = 0;
55   int status = GSL_CONTINUE;
56   status = gsl_root_fsolver_set(solver,F,initLower,initUpper);
57   if(status != GSL_SUCCESS) throw gslError(status,"gsl_root_fsolver_set");
58
59   do{
60     ++iter;
61     // Iterate the root finder
62     status = gsl_root_fsolver_iterate(solver);
63     if(status != GSL_SUCCESS) throw gslError(status,"gsl_root_fsolver_iterate");
64
65     // Test whether the new interval satisfies the specified relative
66     // tolerance. NB: the third parameter to the function specifies the
67     // absolute error, which we take to be the same as the relative tolerance
68     // so that roots close to zero will also converge.
69     status = gsl_root_test_interval(
70       gsl_root_fsolver_x_lower(solver),
71       gsl_root_fsolver_x_upper(solver),
72       rel_tol,rel_tol);
73     if(status != GSL_CONTINUE && status != GSL_SUCCESS)
74       throw gslError(status,"gsl_root_test_interval");
75   } while(status == GSL_CONTINUE && iter < max_iter);
76
77   if(status == GSL_CONTINUE)
78     throw "gslRootSolver::findroot() failed to converge.";
79   if(status != GSL_SUCCESS)
80     throw "gslRootSolver::findroot() failed.";
81
82   return gsl_root_fsolver_root(solver);
83 }
84
85 /*************************************************************************/
86 // 'eqHelper' class
87 // ***************
88 // Provides helper functions for performing optimised equilibration
89 // calculations.
90
91 class eqHelper {
92   public:
93     eqHelper() :
```

```
94        invEr(0.0),invEs(0.0),Hr(0.0),Hs(0.0),
95        RHill(0.0),SHill(0.0),oldCf(0.0),oldTf(0.0) { }
96
97      // Ensure this function is called before any of the calculation functions
98      // are used
99      void initialise(MFLparams* pars) {
100       invEr = 1/pars->Er; invEs = 1/pars->Es;
101       Hr = pars->Hr; Hs = pars->Hs;
102     }
103
104     // Calculating (Cf/Er)^Hr
105     double calcRHill(double Cf) {
106       // Do a quick test to see if Cf is significantly different from the
107       // previous calculation of RHill
108       double difference = Cf - oldCf;
109       if(difference*difference > diffthreshold) {
110   oldCf = Cf;
111   RHill = pow(Cf*invEr,Hr);
112       }
113       return RHill;
114     }
115
116     // Calculating (Tf/Es)^Hs
117     double calcSHill(double Tf) {
118       // Do a quick test to see if Tf is significantly different from the
119       // previous calculation of SHill
120       double difference = Tf - oldTf;
121       if(difference*difference > diffthreshold){
122   oldTf = Tf;
123   SHill = pow(Tf*invEs,Hs);
124       }
125       return SHill;
126     }
127
128   private:
129     // Store local copies of the required parameters; these are set from a
130     // given MFLparams struct by calling 'initialise'
131     double invEr,invEs,Hr,Hs;
132     // Store local copies of the previously calculated results so that the
133     // power calculation can be omitted if the old values are close enough to
134     // the new ones.
135     double RHill,SHill;
136     double oldCf,oldTf;
137 };
```

```
138
139 /*****************************************************************************/
140 // 'equilibrator' classes
141 // **********************
142 // The equilibrator classes specify the functions that are needed to update
143 // the state variables that are dependent on equilibria occuring on much
144 // faster timescales than the simulation step size.
145
146 // Equilibrator for Repressor (Tum−); ignores loss of CI to CI−pR
147 class repressEquil : public equilibrator {
148   public:
149     inline repressEquil(repress* s,MFLparams* p)
150       : equilibrator(s,p),p(p),s(s) {
151   // Initialise the helper class based on the parameters
152   h.initialise(p);
153   // Calculate the concentration of available sites for CI dimer binding
154   // (this parameter is needed only for children that account for loss
155   // of CI dimers through CI−pR binding)
156   ConcCIsites=p->ConcCIsites;
157       }
158
159     // Species calculations
160     // In the simple repressor, the amount of free CI is simply the total
161     // amount of free CI
162     virtual double FreeCI() { return s->Ct; }
163     virtual double PtRate() {
164       // Calculate the production rate from the Pt promoter based on
165       // the amount of repression due to free CI
166       return p->Pt/(1 + h.calcRHill(FreeCI()) );
167     }
168
169     // Equilibrate the state
170     virtual void equilibrate() {
171       s->PtR = PtRate();
172     }
173     virtual void steadystate() {
174       s->Ct = s->PcR / p->Dc;
175       equilibrate();
176       s->Report = p->units_per_conc * s->PtR / p->Dz;
177     }
178
179   protected:
180     double ConcCIsites;
181     MFLparams* p;
```

```
182     eqHelper h; // Create a local copy of the helper class
183
184   private:
185     repress* s;
186 };
187
188 // Equilibrator for Repressor (Tum−); account for loss of CI to CI–pR
189 class repressEquilCRbind : public repressEquil {
190   public:
191     explicit repressEquilCRbind(repress* s,MFLparams* p)
192       : repressEquil(s,p),s(s) {
193       // Initialse the wrapper gsl_function ready for the root solver
194       gslRepressorPoly.function = &repressEquilCRbind::RepressorPolyWrapper;
195       gslRepressorPoly.params = this;
196     }
197
198     // Species calculations
199     virtual double FreeCI() {
200       // When total CI is effectively zero, return zero free CI
201       if(s->Ct <= zeroconc) return zeroconc;
202       else return solver.findroot(&gslRepressorPoly, zeroconc, s->Ct);
203     }
204
205   protected:
206     gslRootSolver solver;
207     gsl_function gslRepressorPoly;
208
209     static double RepressorPolyWrapper(double Cf,void* data){
210       repressEquilCRbind* _this = static_cast<repressEquilCRbind*>(data);
211       return _this->RepressorPoly(Cf);
212     }
213
214     // The polynomial whose roots give FreeCI for the repressor
215     double RepressorPoly(double Cf){
216       return Cf*h.calcRHill(Cf) + (ConcCIsites − s->Ct)*h.calcRHill(Cf) + Cf − s->Ct;
217     }
218
219   private:
220     repress* s;
221 };
222
223 // Equilibrator for MFL (Tum+); ignores loss of CI to CI–pR
224 class mflEquil : public repressEquil {
225   public:
```

```
226     mflEquil(mfl* s,MFLparams* p)
227       : repressEquil(s,p),s(s) {
228       // Initialse gsl_function for root solver
229       gslNoBindPoly.function = &mflEquil::NoBindPolyWrapper;
230       gslNoBindPoly.params = this;
231
232       // Initialise the pre-calculated root solver limits grid:
233       initGrid();
234     }
235
236     // Species calculations
237     virtual double FreeTum() {
238       // When total CI is effectively zero, return total Tum; note that since
239       // total CI thus never goes to 0, Tf will be at least 'diffthreshold'
240       // less than s->Tt
241       if(s->Ct <= zeroconc) return s->Tt;
242       // When total Tum is effectively zero, return zero free Tum
243       if(s->Tt <= zeroconc) return zeroconc;
244
245       // Uncomment to solve without the grid:
246       /* return solveRoot(zeroconc,s->Tt-diffthreshold); */
247
248       // Find bounds for the solver using the pre-computed grid:
249       int i = floor(s->Ct*invStepCt);
250       int j = floor(s->Tt*invStepTt);
251
252
253       double lower, upper;
254
255       if(i < 0 || j < 0) throw "Cannot equilibrate: there are negative concentrations.";
256
257       // Check if we are outside the grid:
258       if(i+1 >= nbins || j+1 >= nbins){
259         // If we are outside default to the max and min possible values:
260         lower = zeroconc;
261         upper = s->Tt-diffthreshold;
262       } else {
263         // Locate the bounds on each corner of the grid surrounding this point:
264         double corners[4];
265         corners[0] = bounds[i][j];
266         corners[1] = bounds[i][j+1];
267         corners[2] = bounds[i+1][j];
268         corners[3] = bounds[i+1][j+1];
269
```

```
270         // Find the minimum and maximum values of the grid
271         lower = corners[0];
272         upper = corners[0];
273         for(int k = 1; k<4; ++k){
274           if(lower > corners[k]) lower = corners[k];
275           if(upper < corners[k]) upper = corners[k];
276         }
277
278         // Output for debugging:
279         /* std::cout<<std::endl<<"Bounds: "; */
280         /* std::cout<<bounds[i][j]<<","; */
281         /* std::cout<<bounds[i][j+1]<<","; */
282         /* std::cout<<bounds[i+1][j]<<","; */
283         /* std::cout<<bounds[i+1][j+1]<<std::endl; */
284
285         // Scale slightly to avoid errors when close to a grid edge
286         lower *= static_cast<double>(0.999);
287         upper *= static_cast<double>(1.001);
288
289         if(upper > s->Tt) upper = s->Tt-diffthreshold;
290       }
291
292       // Output for debugging:
293       /* std::cout<<"i: "<<i<<", j: "<<j<<", Tt: "<<s->Tt; */
294       /* std::cout<<", lower: "<<lower<<", upper: "<<upper; */
295
296       double Tf;
297       try{
298         Tf = solveRoot(lower, upper);
299       } catch(gslError err) {
300         std::cerr<<"Error "<<err.error_code<<" from function '";
301         std::cerr<<err.function<<"'"<<std::endl;
302         std::cerr<<"Root function did not straddle 0. Retrying..."<<std::endl;
303         Tf = solveRoot(zeroconc,s->Tt-diffthreshold);
304       }
305
306       // Output for debugging:
307       /* std::cout<<", Tf: "<<Tf<<", prevIter: "<<solver.prevIterations()<<std::endl; */
308
309       return Tf;
310     }
311
312     virtual double FreeCl() {
313       // If the total Tum is effectively zero, return the total Cl
```

```
314       if(s->Tt <= zeroconc) return s->Ct;
315       else {
316     double Tf = FreeTum();
317     return (s->Tt - Tf)/(2*h.calcSHill(Tf));
318       }
319     }
320
321     virtual double CmplxConc(){
322       // If the total Tum is effectively zero, return zero complex
323       if(s->Tt <= zeroconc) return zeroconc;
324       else {
325     double Tf = FreeTum();
326     return (s->Tt - Tf)/2;
327       }
328     }
329
330     // Equilibrate the state
331     virtual void steadystate() {
332       repressEquil::steadystate();
333       // Make a rough guess of total Tum concentration by reducing the maximum
334       // Tum concentration by the pR repression that would occur by total CI
335       // (i.e. ignore sequestration by any Tum).
336       s->Tt = p->Pt/p->Dt / (1+pow(s->Ct/p->Er,p->Hr));
337       // Improve the guess by one iteration (NB: `repressEquil::steadystate()`
338       // calls `equilibrate()`, and hence `FreeTum()`, to set s->PtR)
339       repressEquil::steadystate();
340       s->Tt = s->PtR / p->Dt;
341
342       // std::cout<<"Steadystate params: "<<std::endl;
343       // p.print(std::cout);std::cout<<std::endl;
344     }
345
346   protected:
347     // Cache upper and lower limits for the solver in a 2D grid of size
348     // nbins*nbins. Indexing is [Ct][Tt]. The choice of nbins=200 seems to
349     // balance between time required to build the array versus the number of
350     // iterations required. Observe the following:
351     //  nbins=20: mean iters = 5.3; generation time = 1 ms;
352     //  nbins=200: mean iters = 3; generation time = 120 ms;
353     //  nbins=500: mean iters = 2.5; generation time = 729 ms;
354     static const int nbins = 200;
355     int maxCt, maxTt;
356     double stepCt, stepTt, invStepCt, invStepTt;
357     double bounds[nbins][nbins];
```

```
358
359     gslRootSolver solver;
360     gsl_function gslNoBindPoly;
361
362     static double NoBindPolyWrapper(double Tf,void* data){
363       mflEquil* _this = static_cast<mflEquil*>(data);
364       return _this->NoBindPoly(Tf);
365     }
366
367     // The polynomial whose roots give the FreeTum for the MFL
368     double NoBindPoly(double Tf){
369       double SHill = h.calcSHill(Tf);
370       return Tf*SHill + (2.0*s->Ct - s->Tt)*SHill + Tf - s->Tt;
371     }
372
373     // Make the solver function virtual so that it can be overridden without
374     // needing to change the 'initGrid' or 'FreeTum' functions
375     virtual double solveRoot(double lower, double upper){
376       return solver.findroot(&gslNoBindPoly,lower,upper);
377     }
378
379     virtual void initGrid(){
380       // Calculate the maximum possible deterministic values for Tt and Ct and
381       // multiply by a factor to account for noise:
382       maxTt = 1.5*p->Pt/p->Dt;
383       maxCt = 1.5*p->PcMax/p->Dc;
384
385       // Calculate the step between gridpoints in the array:
386       stepCt = maxCt/static_cast<double>(nbins);
387       stepTt = maxTt/static_cast<double>(nbins);
388       // Calculate the inverse of the step to initialise for grid locating:
389       invStepCt = 1.0/stepCt;
390       invStepTt = 1.0/stepTt;
391
392       // Initiate the limits array:
393       for(int i=0; i<nbins; ++i){
394         s->Ct = i * stepCt;
395         for(int j=0; j<nbins; ++j){
396           s->Tt = j * stepTt;
397           bounds[i][j] = solveRoot(0.0, s->Tt);
398           // Output for debugging:
399           /* std::cout<<i<<'\t'<<j<<'\t'<<bounds[i][j]<<std::endl; */
400         }
401       }
```

```
402     }
403
404   private:
405     mfl* s;
406 };
407
408 // Equilibrator for MFL (Tum+); accounts for loss of CI to CI-pR
409 class mflEquilCRbind : public mflEquil {
410   public:
411     mflEquilCRbind(mfl* s,MFLparams* p) : mflEquil(s,p),s(s) {
412       // Initialse gsl_function for root solver
413       gslFullPoly.function = &mflEquilCRbind::FullPolyWrapper;
414       gslFullPoly.params = this;
415       initGrid();
416     }
417
418   protected:
419     gsl_function gslFullPoly;
420
421     static double FullPolyWrapper(double Tf,void* data){
422       mflEquilCRbind* _this = static_cast<mflEquilCRbind*>(data);
423       return _this->FullPoly(Tf);
424     }
425
426     // The polynomial whose roots give the FreeTum for the MFL
427     double FullPoly(double Tf){
428       double SFactor = NoBindPoly(Tf);
429       double SHill = h.calcSHill(Tf); // This shouldn't call pow()
430       return ConcCIsites*SHill + SFactor *
431   (1 + pow( 2*p->Er*SHill / (s->Tt-Tf) ,p->Hr));
432     }
433
434     // Override the mflEquil root solver function
435     virtual double solveRoot(double lower, double upper){
436       return solver.findroot(&gslFullPoly,lower,upper);
437     }
438
439     virtual void initGrid(){
440       // Re-initialise the array using the new solveRoot function:
441       // TODO this is costly and should not have to be repeated...
442
443       // NB: the MFL CRbind function is trickier since it blows up when Tt
444       // gets close to zero or close to Tf, so we need to handle those cases
445       // separately. The upper limit given to solveRoot must never be equal
```

```
446         // to Tt.
447
448         // No Tum or CI gives us zeroconc of free Tum
449         bounds[0][0] = zeroconc;
450
451         // No CI gives us Tf equal to total Tum, less the diffthreshold
452         for(int j=1; j<nbins; ++j){
453           bounds[0][j] = j * stepTt - diffthreshold;
454         }
455
456         // No Tum gives us zeroconc reduced by the amount of CI
457         s->Tt = zeroconc;
458         for(int i=1; i<nbins; ++i){
459           s->Ct = i * stepCt;
460           bounds[i][0] = solveRoot(0.0,zeroconc-diffthreshold);
461         }
462
463         for(int i=1; i<nbins; ++i){
464           s->Ct = i * stepCt;
465           for(int j=1; j<nbins; ++j){
466             s->Tt = j * stepTt;
467             bounds[i][j] = solveRoot(0.0,s->Tt-diffthreshold);
468             // Output for debugging:
469             /* std::cout<<i<<'\t'<<j<<'\t'<<bounds[i][j]<<std::endl; */
470           }
471         }
472         // Output for debugging:
473         /* std::cout<<std::endl<<"MFL CRbind grid: "; */
474         /* std::cout<<bounds[0][135]<<","; */
475         /* std::cout<<bounds[0][136]<<","; */
476         /* std::cout<<bounds[1][135]<<","; */
477         /* std::cout<<bounds[1][136]<<std::endl; */
478       }
479
480     private:
481       mfl* s;
482   };
483
484   #endif /* EQUILIBRATION_H */
```

D.3.3 models.h

```
1   ////////////////////////////////////////////////
2   // models.h: Defining MFL models from the library
3
```

```
4  #ifndef MODELS_H
5  #define MODELS_H
6
7  #include "modelinterfaces.h"
8  #include "mflLibrary.h"
9  #include "equilibration.h"
10
11 /* ************************************************************************
12   Model Templates
13 ************************************************************************ */
14
15 /* ************************************************************************ */
16 // 'repressModel' class
17 // ************************
18 // 'repressModel' describes the state of a network where CI represses the pR
19 // (here named Pc) promoter to produce some output reporter activity.
20
21 class repressModel : public model {
22   public:
23
24     virtual void PcAct(double PcA)
25       { s->PcR = p->PcBasal + PcA*(p->PcMax-p->PcBasal); }
26     virtual double PcAct() const
27       { return (s->PcR-p->PcBasal) / (p->PcMax-p->PcBasal); }
28
29     virtual void initLO()
30       { PcAct(0.0); initialise(); }
31     virtual void initHI()
32       { PcAct(1.0); initialise(); }
33
34     virtual void initialise() {
35       s->t = 0.0;
36       s->vol = p->Vinit;
37       eq->steadystate();
38       sim->initialise();
39     }
40
41     // Accessors & mutators
42     void   TotalCI(double CIconc) { s->Ct = CIconc; }
43     double TotalCI() const { return s->Ct; }
44     double PcRate() const { return s->PcR; }
45     double FreeCI() const { return eq->FreeCI(); }
46     double PtRate() const { return s->PtR; }
47
```

```
48      repressModel(repress* s, MFLparams* p, repressEquil* eq, simulator* sim)
49        : model(s,p,eq,sim), s(s), p(p), eq(eq), sim(sim) { }
50
51    private:
52      repress* s;
53      MFLparams* p;
54      repressEquil* eq;
55      simulator* sim;
56  };
57
58  /*****************************************************************************/
59  // 'mflModel' class
60  // ***************
61  // 'mflModel' extends the 'repressModel' by adding in functionality for Tum
62  // and Tum-Cl complex concentrations.
63
64  class mflModel : public repressModel {
65    public:
66
67      // Accessors & mutators
68      void   TotalTum(double TumConc) { s->Tt = TumConc; }
69      double TotalTum() const { return s->Tt; }
70      double FreeTum() const { return eq->FreeTum(); }
71      double CmplxConc() const { return eq->CmplxConc(); }
72
73      mflModel(mfl* s, MFLparams* p, mflEquil* eq, simulator* sim)
74        : repressModel(s,p,eq,sim), s(s), eq(eq) { }
75
76    private:
77      mfl* s;
78      mflEquil* eq;
79  };
80
81  /*****************************************************************************
82    Model Definitions
83  *****************************************************************************/
84
85  /***** Deterministic simulators *****/
86
87  //////////////////////////////////////////////
88  // 'lacDetRepress' model
89  // ************************
90  // Extend basicDetRepress by additionally simulating reporter degradation.
91
```

```
92  class lacDetRepressSim : public simulator {
93    public:
94      lacDetRepressSim(repress* s, MFLparams* p, equilibrator* eq)
95        : simulator(s,p,eq,0.1), s(s), p(p), eq(eq) { }
96
97      void step(double ts) {
98        s->t += ts;
99        s->Ct += (s->PcR - p->Dc*s->Ct)*ts;
100       s->Report += (p->units_per_conc*s->PtR - p->Dz*s->Report)*ts;
101       eq->equilibrate();
102     }
103
104   private: repress* s; MFLparams* p; equilibrator* eq;
105 };
106
107 struct lacDetRepress {
108   lacDetRepress(MFLparams* p)
109     : eq(&s,p), sim(&s,p,&eq), m(&s,p,&eq,&sim) { }
110   repress s; repressEquil eq;
111   lacDetRepressSim sim;
112   repressModel m;
113 };
114
115 struct lacCRbDetRepress {
116   lacCRbDetRepress(MFLparams* p)
117     : eq(&s,p), sim(&s,p,&eq), m(&s,p,&eq,&sim) { }
118   repress s; repressEquilCRbind eq;
119   lacDetRepressSim sim;
120   repressModel m;
121 };
122
123 //////////////////////////////////////////////
124 // 'detMFL' model
125 // ************************
126 // Add in deterministic Tum production & degradation to model the MFL
127
128 class detMFLsim : public simulator {
129   public:
130     detMFLsim(mfl* s, MFLparams* p, equilibrator* eq, const double ts=0.1)
131       : simulator(s, p, eq, ts), s(s), p(p), eq(eq) { }
132
133     void step(double ts) {
134       s->t += ts;
135       s->Ct += (s->PcR - p->Dc*s->Ct)*ts;
```

```
136         s->Tt += (s->PtR - p->Dt*s->Tt)*ts;
137         s->Report += (p->units_per_conc*s->PtR - p->Dz*s->Report)*ts;
138         eq->equilibrate();
139       }
140
141     private: mfl* s; MFLparams* p; equilibrator* eq;
142   };
143
144   struct detMFL {
145     detMFL(MFLparams* p, const double ts=0.1)
146       : eq(&s,p), sim(&s,p,&eq,ts), m(&s,p,&eq,&sim) { }
147     mfl s; mflEquil eq;
148     detMFLsim sim;
149     mflModel m;
150   };
151
152   struct detCRbMFL {
153     detCRbMFL(MFLparams* p)
154       : eq(&s,p), sim(&s,p,&eq), m(&s,p,&eq,&sim) { }
155     mfl s; mflEquilCRbind eq;
156     detMFLsim sim;
157     mflModel m;
158   };
159
160   /***** Stochastic simulators *****/
161
162   ////////////////////////////////////////////
163   // 'stochDiscreteRepress' model
164   // ************************
165   // Hybrid stochastic/deterministic simulation of simple pR repression, where
166   // discrete and continous representations of the model state are
167   // simultaneously tracked. This model additionally simulates growth by a
168   // linear increase in cell volume with periodic and cell division events
169   // (binomial partitioning).
170
171   class stochDiscreteRepressSim : public stochSimulator<repress> {
172     public:
173       stochDiscreteRepressSim(repress* s, MFLparams* p,
174           equilibrator* eq,gsl_rng* r)
175         : stochSimulator<repress>(s,p,eq,r), s(s), p(p), eq(eq)
176       {
177         // Partition molecules at cell division
178         events.push_back(new cellDivisionRepress(s,p,r));
179
```

```
180       // Note that for truly discrete production, molecules should be added as
181       // counts (i.e., not added as concentrations that have a volume
182       // dependence). So use discrete versions of each stochastic event.
183
184       // Discrete production of CI
185       events.push_back(new stochPcDiscrete(s,p,r));
186       // Discrete production of reporter
187       events.push_back(new stochPtReportDiscrete(s,p,r));
188     }
189
190   protected:
191     double DtDiff;
192     void step_detrm(double ts) {
193       // Fast reactions are handled by equilibrator, so the only deterministic
194       // step is a linear increase in volume:
195       s->vol += p->Vrate*ts;
196       // Note that the Volume class then automatically updates all dependent
197       // Variables
198     }
199
200   private: repress* s; MFLparams* p; equilibrator* eq;
201 };
202
203 struct stochDiscreteRepress {
204   stochDiscreteRepress(MFLparams* p, gsl_rng* r)
205     : eq(&s,p), sim(&s,p,&eq,r), m(&s,p,&eq,&sim) { }
206   repress s; repressEquil eq;
207   stochDiscreteRepressSim sim;
208   repressModel m;
209 };
210
211 ////////////////////////////////////////////////
212 // 'stochDiscreteMFL' model
213 // ************************
214 // Hybrid stochastic/deterministic simulation of the MFL where discrete and
215 // continous representations of the model state are simulaneously tracked.
216 // This model additionally simulates growth by volume increases and cell
217 // division (binomial partitioning).
218
219 class stochDiscreteMFLsim : public stochSimulator<mfl> {
220   public:
221     stochDiscreteMFLsim(mfl* s,MFLparams* p,equilibrator* eq,gsl_rng* r)
222       : stochSimulator<mfl>(s,p,eq,r),s(s),p(p),eq(eq)
223     {
```

```
224         // Partition molecules at cell division
225         events.push_back(new cellDivisionMFL(s,p,r));
226
227         // Note that for truly discrete production, molecules should be added as
228         // counts (i.e., not added as concentrations that have a volume
229         // dependence). So use discrete versions of each stochastic event.
230
231         // Discrete production of CI
232         events.push_back(new stochPcDiscrete(s,p,r));
233         // Discrete production of Tum
234         events.push_back(new stochPtTumDiscrete(s,p,r));
235         // Discrete production of reporter
236         events.push_back(new stochPtReportDiscrete(s,p,r));
237
238         // The difference between dilution rate and rate of loss of Tum:
239         DtDiff = -(p->Dt - p->Dc); // Will be positive for 'fittedparams'
240       }
241
242     protected:
243       double DtDiff;
244       void step_detrm(double ts) {
245         // Fast reactions are handled by equilibrator, so the only deterministic
246         // step is a linear increase in volume:
247         s->vol += p->Vrate*ts;
248         // Note that the Volume class then automatically updates all dependent
249         // Variables
250
251         // Since Tum is allowed an alternative degradation (dilution) rate,
252         // which was fitted to a rate much slower than that caused by cell
253         // division, additionally model deterministic 'degradation' of Tum using
254         // the difference between Dt and Dil (NB: this effectively makes for
255         // exponential growth in this case since the difference should be
256         // negative, giving a positive coefficient in the exponent).
257         s->Tt += DtDiff*s->Tt*ts;
258       }
259
260     private: mfl* s; MFLparams* p; equilibrator* eq;
261   };
262
263   struct stochDiscreteMFL {
264     stochDiscreteMFL(MFLparams* p,gsl_rng* r)
265       : eq(&s,p), sim(&s,p,&eq,r), m(&s,p,&eq,&sim) { }
266     mfl s; mflEquil eq;
267     stochDiscreteMFLsim sim;
```

```
268   mflModel m;
269 };
270
271 #endif /* MODELS_H */
```

D.4 Running the simulator

```
1  /* ************************************************************ */
2  // timecourse.cpp
3  // Simulate and return a single time course for various models
4  /* ************************************************************ */
5
6  #include <Rcpp.h>
7  #include "pch.h"
8  #include "modelinterfaces.h"
9  #include "models.h"
10
11 using namespace std;
12 using namespace Rcpp;
13
14 namespace timecourseNamespace {
15
16   // Container for storing repressor model output
17   struct repressArray {
18     vector<double> time;
19     vector<double> PcR;
20     vector<double> Ct;
21     vector<double> Cf;
22     vector<double> PtR;
23     vector<double> Report;
24   };
25
26   // Container for storing MFL model output
27   struct mflArray {
28     vector<double> time;
29     vector<double> PcR;
30     vector<double> Ct;
31     vector<double> Cf;
32     vector<double> PtR;
33     vector<double> Report;
34     vector<double> Tt;
35     vector<double> Tf;
36   };
37
38   // Sample a repress model by appending data to a repressArray
```

```
39  void tcSample(repressArray& out, repressModel& model){
40    out.time.push_back(model.time());
41    out.PcR.push_back(model.PcRate());
42    out.Ct.push_back(model.TotalCI());
43    out.Cf.push_back(model.FreeCI());
44    out.PtR.push_back(model.PtRate());
45    out.Report.push_back(model.activity());
46  }
47
48  // Sample a MFL model by appending data to a mflArray
49  void tcSample(mflArray& out, mflModel& model){
50    out.time.push_back(model.time());
51    out.PcR.push_back(model.PcRate());
52    out.Ct.push_back(model.TotalCI());
53    out.Cf.push_back(model.FreeCI());
54    out.PtR.push_back(model.PtRate());
55    out.Report.push_back(model.activity());
56    out.Tt.push_back(model.TotalTum());
57    out.Tf.push_back(model.FreeTum());
58  }
59
60  // Define function to convert repressArray to DataFrame:
61  DataFrame RcppArray(repressArray& a){
62    return DataFrame::create(
63        Named("time")=a.time,
64        Named("PcR")=a.PcR, Named("PtR")=a.PtR,
65        Named("Ct")=a.Ct, Named("Cf")=a.Cf,
66        Named("Report")=a.Report);
67  }
68
69  // Define function to convert mflArray to DataFrame:
70  DataFrame RcppArray(mflArray& a){
71    return DataFrame::create(
72        Named("time")=a.time,
73        Named("PcR")=a.PcR, Named("PtR")=a.PtR,
74        Named("Ct")=a.Ct, Named("Cf")=a.Cf,
75        Named("Report")=a.Report,
76        Named("Tt")=a.Tt, Named("Tf")=a.Tf);
77  }
78
79  const uint samplingrate = 20;
80
81  /***** Sample states over the course of a simulation *****/
82  template <typename Array, typename Model>
```

```
83      void assay(Array& out, Model& model, bool lo, double activation,
84          double flushtime, double ontime, double assaytime) {
85
86        // Initialise the model
87        if(lo) model.initLO();
88        else model.initHI();
89
90        // Equilibrate/randomize starting positions by simulating for flushtime
91        // minutes (should be around two generations):
92        model.simulate(flushtime);
93        model.time(0.0); // Reset the time but keep the randomised IC
94
95        // Output the initial values:
96        tcSample(out, model);
97
98        // Step through the O/N in defined step sizes using the simulate function:
99        uint nSamples = samplingrate*static_cast<uint>(ontime);
100       double stepsize = ontime/static_cast<double>(nSamples);
101       for(uint i = 0; i<nSamples; ++i){
102         model.simulate(stepsize); // Simulate for a single step
103         tcSample(out, model); // Sample after each time step
104       }
105
106       // Set the assay activation level:
107       model.PcAct(activation);
108
109       // Step through the assay in defined step sizes using the simulate function:
110       nSamples = samplingrate*static_cast<uint>(assaytime);
111       stepsize = assaytime/static_cast<double>(nSamples);
112       for(uint i = 0; i<nSamples; ++i){
113         model.simulate(stepsize); // Simulate for a single step
114         tcSample(out, model); // Sample after each time step
115       }
116     }
117
118   RcppExport SEXP timecourse(SEXP raw_act, SEXP raw_on, SEXP raw_model,
119       SEXP raw_parset, SEXP raw_params, SEXP raw_ontime, SEXP raw_assaytime,
120       SEXP raw_seed, SEXP raw_debug) {
121     try {
122
123       // Determine if we should output any debug information:
124       bool debug = as<bool>(LogicalVector(raw_debug));
125
126       // Construct Rcpp objects from the R arguments
```

```
127     NumericVector R_act(raw_act);
128     CharacterVector R_on(raw_on);
129     CharacterVector R_model(raw_model);
130     CharacterVector R_parset(raw_parset);
131     List R_params(raw_params);
132     NumericVector R_ontime(raw_ontime);
133     NumericVector R_assaytime(raw_assaytime);
134
135     double activation = as<double>(R_act);
136     bool lo = R_on[0] == "LO";
137     double assaytime = as<double>(R_assaytime);
138     double ontime = as<double>(R_ontime);
139
140     // Convert to C++ parameters...
141     double PcBS = as<double>(R_params["PcBS"]);
142     double PtBS = as<double>(R_params["PtBS"]);
143     double PzBS = as<double>(R_params["PzBS"]);
144
145     // Initialise parameter sets
146     mflFittedParams fittedparams(PcBS, PtBS, PzBS);
147     mflManParams manparams(PcBS, PtBS, PzBS);
148
149     // Parameter reference variable
150     MFLparams* params;
151     params = &fittedparams; // Default parameter set
152     if(R_parset[0] == "fitted") params = &fittedparams;
153     if(R_parset[0] == "manual") params = &manparams;
154
155     if(debug){
156       Rprintf("Simulating with:\n");
157       Rprintf("ontime = %f\nassaytime = %f\nactivation = %f\n",
158           ontime, assaytime, activation);
159
160       // Also output the parameters being used:
161       stringstream parstring;
162       params->print(parstring);
163       Rprintf("%s parameters:\n%s",
164           as<string>(R_parset[0]).c_str(),
165           parstring.str().c_str());
166     }
167
168     /*** Shared initialisation code ***/
169
170     // Create a random number generator
```

```
171         // Taus is the fastest (but least 'nice') method
172         gsl_rng * r = gsl_rng_alloc(gsl_rng_mt19937);
173
174         // Since this may be called repeatedly from R very quickly, cannot use
175         // the system time to set the seed:
176         unsigned long int rng_seed = as<unsigned long int>(IntegerVector(raw_seed));
177         gsl_rng_set(r, rng_seed);
178
179         // Since cell division is taken to be periodic, randomise the length of
180         // the flush time uniformly over the doubling time, so that cells start
181         // at random points in the cycle:
182         double flushtime = 60 + gsl_rng_uniform_pos(r)*params->Tdbl;
183
184         // Create an empty DataFrame to be returned after simulations are complete:
185         DataFrame outdf;
186
187         if(R_model[0] == "DR"){
188           repressArray outarray;
189           lacDetRepress DR(params);
190           assay(outarray, DR.m, lo, activation, flushtime, ontime, assaytime);
191           outdf = RcppArray(outarray);
192         }
193
194         if(R_model[0] == "DM"){
195           mflArray outarray;
196           detMFL DM(params);
197           assay(outarray, DM.m, lo, activation, flushtime, ontime, assaytime);
198           outdf = RcppArray(outarray);
199         }
200
201         if(R_model[0] == "SR"){
202           repressArray outarray;
203           stochDiscreteRepress SR(params, r);
204           assay(outarray, SR.m, lo, activation, flushtime, ontime, assaytime);
205           outdf = RcppArray(outarray);
206         }
207
208         if(R_model[0] == "SM"){
209           mflArray outarray;
210           stochDiscreteMFL SM(params, r);
211           assay(outarray, SM.m, lo, activation, flushtime, ontime, assaytime);
212           outdf = RcppArray(outarray);
213         }
214
```

```
215         ////////////////////////////////////////////////
216         // Clean up before exiting...
217
218         // Free up memory used by random number generator
219         gsl_rng_free(r);
220
221         // Return the DataFrame to R (filled or not)
222         return(outdf);
223
224       } catch( std::exception &ex ) {
225         forward_exception_to_r( ex );
226       } catch(...) {
227         ::Rf_error( "c++ exception (unknown reason)" );
228       }
229       return R_NilValue; // -Wall
230     }
231
232 };
```

```
1  # Define an R interface function to check and curate arguments before handing
2  # over to the C++ 'timecourse' routine:
3  timecourse <- function(activation=0.2, on=c('LO','HI'),
4                         model=c('SM','SR','DM','DR'),
5                         parset=c('fitted','manual'),
6                         params=list(PcBS=100, PtBS=100, PzBS=10),
7                         ontime=60, assaytime=250, debug=FALSE){
8    # Match 'on', 'model' and 'parset', to the formals of this function:
9    on <- match.arg(on)
10   model <- match.arg(model)
11   parset <- match.arg(parset)
12
13   # Make sure that we have all of the params as specified in the default
14   # argument list; if some are missing, then set those to their defaults:
15   requiredParams <- formals(timecourse)$params
16   missingParams <-
17     names(requiredParams)[!(names(requiredParams) %in% names(params))]
18   params[missingParams] <- requiredParams[missingParams]
19   if(debug) print(params)
20
21   # Ensure the correct type for all other arguments:
22   if(!(is.numeric(activation) || is.numeric(ontime) || is.numeric(assaytime)))
23     stop('"activation", "ontime" and "assaytime" must all be numeric.')
24
25   ontime <- abs(ontime) # Make sure ontime is positive
26   assaytime <- abs(assaytime) # Make sure assaytime is positive
```

```
27
28    if(activation<0 || activation>1)
29      stop('"activation" must be in [0,1]')
30
31    # Choose a new seed for this instance:
32    seed <- sample.int(.Machine$integer.max, 1)
33
34    .Call("timecourse", PACKAGE='stochlib',
35          activation, on, model, parset, params, ontime, assaytime, seed, debug)
36  }
37
38  # Run a complete MFL experiment multiple times:
39  tcExpt <- function(activation=0.2, model=c('M','R'),
40                     parset=c('fitted','manual'),
41                     params=list(PcBS=100, PtBS=100, PzBS=10),
42                     n=10, ontime=60, assaytime=250, debug=FALSE,
43                     col=c('blue','red'), lty='solid'){
44    model <- match.arg(model)
45    dataset <- list()
46    cols <- rep(col, each=trunc(4/length(col)))
47    cols <- lapply(cols, function(x) rgb2hsv(col2rgb(x)))
48
49    tctags <- paste0(c('sl','sh','dl','dh'), model)
50
51    # Run multiple stochastic simulations
52    for(i in 1:n){
53      for(j in 1:2){
54        dataset[[tctags[j]]]$data[[i]] <-
55          timecourse(activation, on=if(j==1) 'LO' else 'HI',
56                     model=paste0('S',model), parset=parset, params=params,
57                     ontime=ontime, assaytime=assaytime, debug=debug)
58        dataset[[tctags[j]]]$cols[[i]] <- hsv(cols[[j]][1], i/n, cols[[j]][3])
59        dataset[[tctags[j]]]$ltys[[i]] <- lty[1+(j-1)%%length(lty)]
60      }
61    }
62
63    # Run single deterministic simulations
64    for(j in 3:4){
65      dataset[[tctags[j]]]$data[[1]] <-
66        timecourse(activation, on=if(j==3) 'LO' else 'HI',
67                   model=paste0('D',model), parset=parset, params=params,
68                   ontime=ontime, assaytime=assaytime, debug=debug)
69      dataset[[tctags[j]]]$cols[[1]] <- hsv(cols[[j]][1], cols[[j]][2], cols[[j]][3])
70      dataset[[tctags[j]]]$ltys[[1]] <- lty[1+(j-1)%%length(lty)]
```

```
71    }
72
73    structure(dataset, title=model)
74  }
```

D.5 Adding experimental noise to stochastic simulations

In order to best replicate the experimental data, background noise was added to the simulated results. Experimental noise was simulated by randomly sampling from a normal distribution with a mean (offset) and standard deviation chosen so that the distribution would match the spread of the fully repressed control samples. This background noise was added to the stochastic simulation data before being analysed like the flow cytometry samples were.

The following code block shows how this was achieved in R using the R library described in this appendix. The `assayocc` and `assaydet` functions in the listing are similar to the time course simulation code shown in the previous section, except that they return a large collection of activities at a specified time point for stochastic or deterministic simulations respectively. The `'DR'` and `'DM'` models specified in calls to the `assayocc` function choose between repressor and MFL stochastic models of the (Discrete) hybrid stochastic/deterministic model.

After adding in noise, the data is then analysed using code taken from the flow cytometry library that was customised only to handle the different format of the data structures shown below. These steps included (1) Logicle transformation of all data, (2) fitting the control simulations with skew-*t* distributions, (3) training the constrained skew-*t* model using this data, and (4) fitting the MFL simulations with a constrained skew-*t* mixture model.

```
1   library(stochlib) # The stochastic simulation library
2
3   ### EXCERPTED: customised flow cytometry analysis code ###
4
5   # Set the number of simulations to run
6   Nlarge <- 3e4
7   # Simulation times are 3, 4 and 5 hours
8   simtimes <- list(short=180, medium=240, long=300)
9
10  # Choose the levels of induction for stochastic simulation...
11  indLevels <- seq(0,1,length.out=24)
12  # ...and for deterministic simulation
13  indLevels.det <- seq(0,1,length.out=200)
14
15  # Define the background noise that will be generated to simulate the
16  # experimental background:
```

```
17  bgdPars <- list(offset=1674*1000/5861, bgsd=1254*1000/5861)
18
19  # Set parameters as per the other thesis figures:
20  manpars <- list(PcBS=300,PtBS=3000,PzBS=100)
21
22  # Start by building a model of the control distributions: {{{
23
24  # Standard deterministic simulations
25  SRdet.man <-
26    lapply(simtimes, function(simtime) {
27          assaydet(indLevels.det, model='R', parset='manual',
28                   params=manpars, assaytime=simtime) })
29
30  # Hybrid stochastic simulations
31  SRocc.man <-
32    lapply(simtimes, function(simtime) {
33          assayocc(indLevels, model='DR', nsamples=Nlarge,
34                   parset='manual', params=manpars, assaytime=simtime) })
35
36  PcR.man <- sapply(SRocc.man[[1]]$histlo, function(x) x$PcR[[1]])
37
38  # Add background noise
39  SRocc.bgd.man <-
40    lapply(SRocc.man, function(assay)
41          lapply(assay, function(x)
42                lapply(x, function(sim) {
43                      with(bgdPars, sim$Report + rnorm(Nlarge,offset,bgsd))
44                      }))))
45
46  # Determine Logicle transformation parameters
47  negref.man <-
48    lapply(SRocc.bgd.man, function(assay)
49          c(Reduce(c, assay$histlo, numeric()),
50            Reduce(c, assay$histhi, numeric())) )
51  negref.man <-
52    lapply(negref.man, function(ng) unname(quantile(ng[ng<=0], probs=0.05)))
53  logicleW.man <-
54    lapply(negref.man, function(ng) (4.5-log10(262144/abs(ng)))/2)
55
56  # Do the Logicle transformation
57  SRocc.bgd.log.man <-
58    mapply(function(assay, Wpar) {
59          lapply(assay, function(on) {
60                cat('\n')
```

```
                    lapply(on, function(ffdata){
                            cat('.'); invLogicle(ffdata, Wpar)
                            }) })
            }, SRocc.bgd.man, logicleW.man, SIMPLIFY=FALSE)

SRocc.fits.man <-
  lapply(SRocc.bgd.log.man, function(assay)
          skewt.unifit(with(assay, c(histlo, histhi))) )

SRocc.predictor.man <-
  lapply(SRocc.fits.man, function(fits) skewt.predictor(fits))

# END control distributions }}}

# Now fit the constrained skew t model to simulated MFL distributions: {{{

# Standard deterministic simulations
SMdet.man <-
  lapply(simtimes, function(simtime) {
          assaydet(indLevels.det, model='M', parset='manual',
                    params=manpars, assaytime=simtime) })

# Hybrid stochastic simulations
SMocc.man <-
  lapply(simtimes, function(simtime) {
          assayocc(indLevels, model='DM', nsamples=Nlarge,
                    parset='manual', params=manpars, assaytime=simtime) })

# Pc induction levels are the same as PcR.man
all(PcR.man == sapply(SMocc.man[[1]]$histlo, function(x) x$PcR[[1]])) # TRUE

# Add background noise
SMocc.bgd.man <-
  lapply(SMocc.man, function(assay)
          lapply(assay, function(x)
                  lapply(x, function(sim) {
                          with(bgdPars, sim$Report + rnorm(Nlarge,offset,bgsd))
                          })))

# Use the same Logicle transformation parameters as the repressor controls

# Do the Logicle transformation
SMocc.bgd.log.man <-
  mapply(function(assay, Wpar) {
```

```
105             lapply(assay, function(on) {
106                     cat('\n')
107                     lapply(on, function(ffdata) {
108                             cat('.'); invLogicle(ffdata, Wpar)
109                             }) })
110             }, SMocc.bgd.man, logicleW.man, SIMPLIFY=FALSE)
111
112     # Fit mixture models (unimodal and bimodal) to the data:
113     # NB: the following was re-run with even more partitions in the initial
114     # parameter search in the 'bimodal.skewt.em' function
115     SMocc.mixfits.man <-
116       mcmapply(function(assay, predictor)
117                 lapply(assay, ModalityAnalysis, predictor=predictor),
118                 SMocc.bgd.log.man, SRocc.predictor.man, SIMPLIFY=FALSE)
119
120     SMocc.summary.man <-
121       mcmapply(function(assayfits, predictor, Wpar)
122                 lapply(assayfits, summary, W=Wpar, predictor=predictor),
123                 SMocc.mixfits.man, SRocc.predictor.man, logicleW.man, SIMPLIFY=FALSE)
124
125     SMocc.summary.man <-
126       lapply(SMocc.summary.man, function(equil)
127             lapply(equil, function(x) within(x, bimodal <- valuediff < -100)))
128
129     # END MFL distributions }}}
```

Bibliography

Ackers, G. K., Johnson, A. D., and Shea, M. A. (1982). Quantitative model for gene regulation by lambda phage repressor. *Proc. Nat. Acad. Sci. U.S.A.*, 79(4):1129–33.

Agapakis, C. M. and Silver, P. A. (2009). Synthetic biology: exploring and exploiting genetic modularity through the design of novel biological networks. *Mol. Biosyst.*, 5(7):704–13.

Alfonsi, A., Cancès, E., Turinici, G., Di Ventura, B., and Huisinga, W. (2005). Adaptive simulation of hybrid stochastic and deterministic models for biochemical systems. In *ESAIM: Proceedings*, volume 14, pages 1–13. EDP Sciences.

Alon, U. (2003). Biological networks: the tinkerer as an engineer. *Science*, 301(5641):1866–7.

Alon, U. (2007). Network motifs: theory and experimental approaches. *Nat. Rev. Genet.*, 8(6):450–461.

Alsing, A., Pedersen, M., Sneppen, K., and Hammer, K. (2011). Key players in the genetic switch of bacteriophage tp901-1. *Biophys. J.*, 100(2):313–21.

Anand, R., Rai, N., and Thattai, M. (2011). Promoter reliability in modular transcriptional networks. *Methods Enzymol.*, 497:31–49.

Anderson, J. C., Clarke, E. J., Arkin, A. P., and Voigt, C. A. (2006). Environmentally controlled invasion of cancer cells by engineered bacteria. *J. Mol. Biol.*, 355(4):619–27.

Andrianantoandro, E., Basu, S., Karig, D. K., and Weiss, R. (2006). Synthetic biology: new engineering rules for an emerging discipline. *Molecular Systems Biology*, 2(28).

Arkin, A., Ross, J., and McAdams, H. H. (1998). Stochastic kinetic analysis of developmental pathway bifurcation in phage λ-infected *Escherichia coli* cells. *Genetics*, 149(4):1633–48.

Atkinson, M. R., Savageau, M. A., Myers, J. T., and Ninfa, A. J. (2003). Development of genetic circuitry exhibiting toggle switch or oscillatory behavior in *Escherichia coli*. *Cell*, 113(5):597–607.

Atsumi, S. and Little, J. W. (2006). A synthetic phage lambda regulatory circuit. *Proc. Nat. Acad. Sci. U.S.A.*, 103(50):19045–19050.

Aurell, E. and Sneppen, K. (2002). Epigenetics as a first exit problem. *Phys. Rev. Lett.*, 88(4):048101.

Azzalini, A. and Capitanio, A. (2003). Distributions generated by perturbation of symmetry with emphasis on a multivariate skew *t*-distribution. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 65(2):367–389.

Baetz, K. and Andrews, B. (1999). Regulation of cell cycle transcription factor Swi4 through auto-inhibition of DNA binding. *Mol. Cell Biol.*, 19(10):6729–6741.

Balaban, N. Q., Merrin, J., Chait, R., Kowalik, L., and Leibler, S. (2004). Bacterial persistence as a phenotypic switch. *Science*, 305(5690):1622–5.

Barabasi, A.-L. and Oltvai, Z. N. (2004). Network biology: understanding the cell's functional organization. *Nat Rev Genet*, 5(2):101–113.

Bashashati, A. and Brinkman, R. R. (2009). A survey of flow cytometry data analysis methods. *Adv. Bioinformatics*, page 584603.

Basso, R. M., Lachos, V. H., Cabral, C. R. B., and Ghosh, P. (2010). Robust mixture modeling based on scale mixtures of skew-normal distributions. *Computational Statistics & Data Analysis*, 54(12):2926–2941.

Battesti, A., Majdalani, N., and Gottesman, S. (2011). The RpoS-mediated general stress response in *Escherichia coli*. *Annu. Rev. Microbiol.*, 65:189–213.

Bayer, M. E. (1991). Zones of membrane adhesion in the cryofixed envelope of *Escherichia coli*. *J. Struct. Biol.*, 107(3):268–80.

Becskei, A., Seraphin, B., and Serrano, L. (2001). Positive feedback in eukaryotic gene networks: cell differentiation by graded to binary response conversion. *EMBO J.*, 20(10):2528–2535.

Becskei, A. and Serrano, L. (2000). Engineering stability in gene networks by autoregulation. *Nature*, 405(6786):590–593.

Belasco, J. G. (2010). All things must pass: contrasts and commonalities in eukaryotic and bacterial mRNA decay. *Nat. Rev. Mol. Cell Biol.*, 11(7):467–78.

Benner, S. A. and Sismour, A. M. (2005). Synthetic biology. *Nat. Rev. Genet.*, 6(7):533–543.

Bernstein, J. A., Khodursky, A. B., Lin, P.-H., Lin-Chao, S., and Cohen, S. N. (2002). Global analysis of mRNA decay and abundance in *Escherichia coli* at single-gene resolution using two-color fluorescent DNA microarrays. *Proc. Nat. Acad. Sci. U.S.A.*, 99(15):9697–702.

Bintu, L., Buchler, N. E., Garcia, H. G., Gerland, U., Hwa, T., Kondev, J., Kuhlman, T., and Phillips, R. (2005). Transcriptional regulation by the numbers: applications. *Curr. Opin. Genet. Dev.*, 15(2):125–35.

Bipatnath, M., Dennis, P. P., and Bremer, H. (1998). Initiation and velocity of chromosome replication in Escherichia coli B/r and K-12. *J. Bacteriol.*, 180(2):265–73.

Box, G. E. P. and Cox, D. R. (1964). An analysis of transformations. *Journal of the Royal Statistical Society. Series B (Methodological)*, 26(2):pp. 211–252.

Brandman, O., Ferrell, Jr, J. E., Li, R., and Meyer, T. (2005). Interlinked fast and slow positive feedback loops drive reliable cell decisions. *Science*, 310(5747):496–8.

Bremer, H. and Dennis, P. P. (1996). Modulation of Chemical Composition and Other Parameters of the Cell by Growth Rate. In Neidhardt, F. C., Curtiss, R., Ingraham, J. L., Lin, E. C. C., Low, K. B., Magasanik, B., Reznikoff, W. S., Riley, M., Schaechter, M., and Umbarger, H. E., editors, *Escherichia coli and Salmonella typhimurium: Cellular and Molecular Biology*, volume 2, chapter 97, pages 1553–1569. American Society for Microbiology, 2nd edition.

Brophy, J. A. N. and Voigt, C. A. (2014). Principles of genetic circuit design. *Nat. Methods*, 11(5):508–20.

Brumby, A. M., Lamont, I., Dodd, I. B., and Egan, J. B. (1996). Defining the SOS operon of coliphage 186. *Virology*, 219(1):105–114.

Buchler, N. E. and Cross, F. R. (2009). Protein sequestration generates a flexible ultrasensitive response in a genetic network. *Mol. Syst. Biol.*, 5:272.

Buchler, N. E. and Louis, M. (2008). Molecular titration and ultrasensitivity in regulatory networks. *J. Mol. Biol.*, 384(5):1106–1119.

Burrill, D. R. and Silver, P. A. (2010). Making cellular memories. *Cell*, 140(1):13–8.

Cağatay, T., Turcotte, M., Elowitz, M. B., Garcia-Ojalvo, J., and Süel, G. M. (2009). Architecture-dependent noise discriminates functionally analogous differentiation circuits. *Cell*, 139(3):512–22.

Cai, L., Friedman, N., and Xie, X. S. (2006). Stochastic protein expression in individual cells at the single molecule level. *Nature*, 440(7082):358–62.

Callen, B. P., Shearwin, K. E., and Egan, J. B. (2004). Transcriptional interference between convergent promoters caused by elongation over the promoter. *Mol. Cell*, 14(5):647–656.

Canton, B., Labno, A., and Endy, D. (2008). Refinement and standardization of synthetic biological parts and devices. *Nat. Biotechnol.*, 26(7):787–93.

Chai, Y., Kolter, R., and Losick, R. (2010a). Reversal of an epigenetic switch governing cell chaining in *Bacillus subtilis* by protein instability. *Mol. Microbiol.*, 78(1):218–29.

Chai, Y., Norman, T., Kolter, R., and Losick, R. (2010b). An epigenetic switch governing daughter cell separation in *Bacillus subtilis*. *Genes Dev.*, 24(8):754–65.

Chang, D.-E., Leung, S., Atkinson, M. R., Reifler, A., Forger, D., and Ninfa, A. J. (2010). Building biological memory by linking positive feedback loops. *Proc. Nat. Acad. Sci. U.S.A.*, 107(1):175–80.

Chen, D. and Arkin, A. P. (2012). Sequestration-based bistability enables tuning of the switching boundaries and design of a latch. *Mol. Syst. Biol.*, 8:620.

Chung, C. T., Niemela, S. L., and Miller, R. H. (1989). One-step preparation of competent Escherichia coli: transformation and storage of bacterial cells in the same solution. *Proc. Nat. Acad. Sci. U.S.A.*, 86(7):2172–5.

Crameri, A., Whitehorn, E. A., Tate, E., and Stemmer, W. P. (1996). Improved green fluorescent protein by molecular evolution using DNA shuffling. *Nat. Biotechnol.*, 14(3):315–9.

Crooks, M. (2006). *Transcriptional Interference Between the Switch Promoters of 186*. Honours thesis, University of Adelaide.

Cui, L., Murchland, I., Shearwin, K. E., and Dodd, I. B. (2013). Enhancer-like long-range transcriptional activation by λ CI-mediated DNA looping. *Proc. Nat. Acad. Sci. U.S.A.*, 110(8):2922–7.

Del Vecchio, D., Ninfa, A. J., and Sontag, E. D. (2008). Modular cell biology: retroactivity and insulation. *Mol. Syst. Biol.*, 4:161.

Dodd, I. B. and Egan, J. B. (1996). DNA binding by the coliphage 186 repressor protein CI. *J. Biol. Chem.*, 271(19):11532–11540.

Dodd, I. B. and Egan, J. B. (2002). Action at a distance in CI repressor regulation of the bacteriophage 186 genetic switch. *Mol. Microbiol.*, 45(3):697–710.

Dodd, I. B., Kalionis, B., and Egan, J. B. (1990). Control of gene expression in the temperate coliphage 186. VIII. Control of lysis and lysogeny by a transcriptional switch involving face-to-face promoters. *J. Mol. Biol.*, 214(1):27–37.

Dodd, I. B., Micheelsen, M. A., Sneppen, K., and Thon, G. (2007a). Theoretical analysis of epigenetic cell memory by nucleosome modification. *Cell*, 129(4):813–822.

Dodd, I. B., Perkins, A. J., Tsemitsidis, D., and Egan, J. B. (2001). Octamerization of λ CI repressor is needed for effective repression of P_{RM} and efficient switching from lysogeny. *Genes Dev.*, 15(22):3013–3022.

Dodd, I. B., Shearwin, K. E., Perkins, A. J., Burr, T., Hochschild, A., and Egan, J. B. (2004). Cooperativity in long-range gene regulation by the lambda CI repressor. *Genes Dev.*, 18(3):344–54.

Dodd, I. B., Shearwin, K. E., and Sneppen, K. (2007b). Modelling transcriptional interference and DNA looping in gene regulation. *J. Mol. Biol.*, 369(5):1200–1213.

Donachie, W. D. and Robinson, A. C. (1987). Cell Division: Parameter Values and the Process. In Neidhardt, F. C., Ingraham, J. L., Low, K. B., Magasanik, B., Schaechter, M., and Umbarger, H. E., editors, *Escherichia coli and Salmonella typhimurium: Cellular and Molecular Biology*, volume 2, chapter 99, pages 1578–1593. American Society for Microbiology.

Dunlop, M. J., Cox, 3rd, R. S., Levine, J. H., Murray, R. M., and Elowitz, M. B. (2008). Regulatory activity revealed by dynamic correlations in gene expression noise. *Nat. Genet.*, 40(12):1493–8.

Elf, J. and Ehrenberg, M. (2003). Fast evaluation of fluctuations in biochemical networks with the linear noise approximation. *Genome Res*, 13(11):2475–84.

Ellis, B., Haaland, P., Hahne, F., Meur, N. L., and Gopalakrishnan, N. (2009a). *flowCore: Basic structures for flow cytometry data*. R package version 1.22.3.

Ellis, T., Wang, X., and Collins, J. J. (2009b). Diversity-based, model-guided construction of synthetic gene networks with predicted functions. *Nat. Biotechnol.*, 27(5):465–71.

Elowitz, M. B. and Leibler, S. (2000). A synthetic oscillatory network of transcriptional regulators. *Nature*, 403(6767):335–338.

Elowitz, M. B., Levine, A. J., Siggia, E. D., and Swain, P. S. (2002). Stochastic gene expression in a single cell. *Science*, 297(5584):1183–6.

Elowitz, M. B., Surette, M. G., Wolf, P. E., Stock, J. B., and Leibler, S. (1999). Protein mobility in the cytoplasm of escherichia coli. *J. Bacteriol.*, 181(1):197–203.

Endy, D. (2005). Foundations for engineering biology. *Nature*, 438(7067):449–453.

Eydgahi, H., Chen, W. W., Muhlich, J. L., Vitkup, D., Tsitsiklis, J. N., and Sorger, P. K. (2013). Properties of cell death models calibrated and compared using Bayesian approaches. *Mol. Syst. Biol.*, 9:644.

Ferrell, J. E. J. (2002). Self-perpetuating states in signal transduction: positive feedback, double-negative feedback and bistability. *Curr. Opin. Cell. Biol.*, 14(2):140–148.

Finak, G., Perez, J.-M., Weng, A., and Gottardo, R. (2010). Optimizing transformations for automated, high throughput analysis of flow cytometry data. *BMC Bioinformatics*, 11:546.

François, P. and Hakim, V. (2005). Core genetic module: the mixed feedback loop. *Phys. Rev. E*, 72(3):031908.

Friedman, N., Cai, L., and Xie, X. S. (2006). Linking stochastic dynamics to population distribution: an analytical framework of gene expression. *Phys. Rev. Lett.*, 97(16):168302.

Gardner, T. S., Cantor, C. R., and Collins, J. J. (2000). Construction of a genetic toggle switch in escherichia coli. *Nature*, 403(6767):339–342.

Gentleman, R. C., Carey, V. J., Bates, D. M., and others (2004). Bioconductor: Open software development for computational biology and bioinformatics. *Genome Biology*, 5:R80. `http://genomebiology.com/2004/5/10/R80`.

Gibson, D. G., Benders, G. A., Andrews-Pfannkoch, C., Denisova, E. A., Baden-Tillson, H., Zaveri, J., Stockwell, T. B., Brownley, A., Thomas, D. W., Algire, M. A., Merryman, C., Young, L., Noskov, V. N., Glass, J. I., Venter, J. C., Hutchison, 3rd, C. A., and Smith, H. O. (2008). Complete chemical synthesis, assembly, and cloning of a mycoplasma genitalium genome. *Science*, 319(5867):1215–20.

Gibson, D. G., Glass, J. I., Lartigue, C., Noskov, V. N., Chuang, R.-Y., Algire, M. A., Benders, G. A., Montague, M. G., Ma, L., Moodie, M. M., Merryman, C., Vashee, S., Krishnakumar, R., Assad-Garcia, N., Andrews-Pfannkoch, C., Denisova, E. A., Young, L., Qi, Z.-Q., Segall-Shapiro, T. H., Calvey, C. H., Parmar, P. P., Hutchison, 3rd, C. A., Smith, H. O., and Venter, J. C. (2010). Creation of a bacterial cell controlled by a chemically synthesized genome. *Science*, 329(5987):52–6.

Gibson, M. and Bruck, J. (2000). Efficient exact stochastic simulation of chemical systems with many species and many channels. *J. Phys. Chem. A*, 104(9):1876–1889.

Gillespie, D. T. (1977). Exact Stochastic Simulation of Coupled Chemical Reactions. *Journal of Physical Chemistry*, 81(25):2340–2361.

Gillespie, D. T. (1992). A rigorous derivation of the chemical master equation. *Physica A: Statistical Mechanics and its Applications*, 188(1):404–425.

Gillespie, D. T. (2000). The chemical langevin equation. *The Journal of Chemical Physics*, 113(1):297–306.

Giza, D., Singla, P., and Jah, M. (2009). An approach for nonlinear uncertainty propagation: Application to orbital mechanics. In *AIAA Guidance, Navigation, and Control Conference, Chicago IL*, pages 1–19.

Golding, I., Paulsson, J., Zawilski, S. M., and Cox, E. C. (2005). Real-time kinetics of gene activity in individual bacteria. *Cell*, 123(6):1025–36.

Gottesman, S. and Maurizi, M. R. (1992). Regulation by proteolysis: energy-dependent proteases and their targets. *Microbiol. Rev.*, 56(4):592–621.

Graham, L. L., Beveridge, T. J., and Nanninga, N. (1991). Periplasmic space and the concept of the periplasm. *Trends Biochem. Sci.*, 16(9):328–9.

Greber, D., El-Baba, M. D., and Fussenegger, M. (2008). Intronically encoded siRNAs improve dynamic range of mammalian gene regulation systems and toggle switch. *Nucleic Acids Res.*, 36(16):e101.

Grima, R., Thomas, P., and Straube, A. V. (2011). How accurate are the nonlinear chemical Fokker-Planck and chemical Langevin equations? *J. Chem. Phys.*, 135(8):084103.

Guet, C. C., Elowitz, M. B., Hsing, W., and Leibler, S. (2002). Combinatorial synthesis of genetic networks. *Science*, 296(5572):1466–70.

Guido, N. J., Wang, X., Adalsteinsson, D., McMillen, D., Hasty, J., Cantor, C. R., Elston, T. C., and Collins, J. J. (2006). A bottom-up approach to gene regulation. *Nature*, 439(7078):856–860.

Gupta, R. S. (1975). Killing and Lysis of *Escherichia coli* in the Presence of Chloramphenicol: Relation to Cellular Magnesium. *Antimicrob. Agents Chemother.*, 7(6):748–753.

Haldimann, A. and Wanner, B. L. (2001). Conditional-replication, integration, excision, and retrieval plasmid-host systems for gene structure-function studies of bacteria. *J. Bacteriol.*, 183(21):6384–6393.

Hammar, P., Walldén, M., Fange, D., Persson, F., Baltekin, O., Ullman, G., Leroy, P., and Elf, J. (2014). Direct measurement of transcription factor dissociation excludes a simple operator occupancy model for gene regulation. *Nat. Genet.*, 46(4):405–8.

Hartwell, L. H., Hopfield, J. J., Leibler, S., and Murray, A. W. (1999). From molecular to modular cell biology. *Nature*, 402(6761 Suppl):C47–52.

Haseltine, E. L. and Rawlings, J. B. (2002). Approximate simulation of coupled fast and slow reactions for stochastic chemical kinetics. *J. Chem. Phys.*, 117(15):6959–6969.

Hasty, J., Isaacs, F., Dolnik, M., McMillen, D., and Collins, J. (2001). Designer gene networks: Towards fundamental cellular control. *Chaos*, 11(1):207–220.

Hasty, J., McMillen, D., and Collins, J. J. (2002). Engineered gene circuits. *Nature*, 420(6912):224–230.

Henderson, D. A., Boys, R. J., Krishnan, K. J., Lawless, C., and Wilkinson, D. J. (2009). Bayesian emulation and calibration of a stochastic computer model of mitochondrial

DNA deletions in substantia nigra neurons. *Journal of the American Statistical Association*, 104(485).

Herzenberg, L. A., Tung, J., Moore, W. A., Herzenberg, L. A., and Parks, D. R. (2006). Interpreting flow cytometry data: a guide for the perplexed. *Nat. Immunol.*, 7(7):681–5.

Hindmarsh, A. (1983). ODEPACK, A Systematized Collection of ODE Solvers. In Stepleman, R. A., editor, *IMACS Transactions on Scientific Computation*, volume 1, pages 55–64. Scientific Computing, North-Holland, Amsterdam.

Hlavacek, W. S. and Savageau, M. A. (1996). Rules for coupled expression of regulator and effector genes in inducible circuits. *J. Mol. Biol.*, 255(1):121–39.

Ho, H. J., Lin, T. I., Chang, H. H., Haase, S. B., Huang, S., and Pyne, S. (2012). Parametric modeling of cellular state transitions as measured with flow cytometry. *BMC Bioinformatics*, 13 Suppl 5:S5.

Hooshangi, S., Thiberge, S., and Weiss, R. (2005). Ultrasensitivity and noise propagation in a synthetic transcriptional cascade. *Proc. Nat. Acad. Sci. U.S.A.*, 102(10):3581–3586.

Huh, J. H., Kittleson, J. T., Arkin, A. P., and Anderson, J. C. (2013). Modular design of a synthetic payload delivery device. *ACS Synth. Biol.*, 2(8):418–24.

Isaacs, F. J., Hasty, J., Cantor, C. R., and Collins, J. J. (2003). Prediction and measurement of an autoregulatory genetic module. *Proc. Nat. Acad. Sci. U.S.A.*, 100(13):7714–7719.

Jayanthi, S., Nilgiriwala, K. S., and Del Vecchio, D. (2013). Retroactivity controls the temporal dynamics of gene transcription. *ACS Synth. Biol.*, 2(8):431–41.

Joyce, A. R. and Palsson, B. Ø. (2006). The model organism as a system: integrating 'omics' data sets. *Nat. Rev. Mol. Cell Biol.*, 7(3):198–210.

Kaern, M., Blake, W. J., and Collins, J. J. (2003). The engineering of gene regulatory networks. *Annu Rev Biomed Eng*, 5:179–206.

Kaern, M., Elston, T. C., Blake, W. J., and Collins, J. J. (2005). Stochasticity in gene expression: from theories to phenotypes. *Nat Rev Genet*, 6(6):451–464.

Kalionis, B., Dodd, I. B., and Egan, J. B. (1986). Control of gene expression in the P2-related template coliphages. III. DNA sequence of the major control region of phage 186. *J. Mol. Biol.*, 191(2):199–209.

Karlebach, G. and Shamir, R. (2008). Modelling and analysis of gene regulatory networks. *Nat. Rev. Mol. Cell Biol.*, 9(10):770–80.

Keller, A. D. (1995). Model genetic circuits encoding autoregulatory transcription factors. *J. Theor. Biol.*, 172(2):169–85.

Kepler, T. B. and Elston, T. C. (2001). Stochasticity in transcriptional regulation: origins, consequences, and mathematical representations. *Biophys J*, 81(6):3116–36.

Khalil, A. S. and Collins, J. J. (2010). Synthetic biology: applications come of age. *Nat. Rev. Genet.*, 11(5):367–79.

Kiehl, T. R., Mattheyses, R. M., and Simmons, M. K. (2004). Hybrid simulation of cellular behavior. *Bioinformatics*, 20(3):316–22.

Kim, J., White, K. S., and Winfree, E. (2006). Construction of an in vitro bistable circuit from synthetic transcriptional switches. *Molecular Systems Biology*, 2(68).

Kittisopikul, M. and Süel, G. M. (2010). Biological role of noise encoded in a genetic network motif. *Proc. Nat. Acad. Sci. U.S.A.*, 107(30):13300–5.

Klumpp, S., Zhang, Z., and Hwa, T. (2009). Growth rate-dependent global effects on gene expression in bacteria. *Cell*, 139(7):1366–75.

Knijnenburg, T. A., Roda, O., Wan, Y., Nolan, G. P., Aitchison, J. D., and Shmulevich, I. (2011). A regression model approach to enable cell morphology correction in high-throughput flow cytometry. *Mol. Syst. Biol.*, 7:531.

Kobayashi, H., Kaern, M., Araki, M., Chung, K., Gardner, T. S., Cantor, C. R., and Collins, J. J. (2004). Programmable cells: interfacing natural and engineered gene networks. *Proc. Nat. Acad. Sci. U.S.A.*, 101(22):8414–9.

Koch, A. L., Robertson, B. R., and Button, D. K. (1996). Deduction of the cell volume and mass from forward scatter intensity of bacteria analyzed by flow cytometry. *Journal of Microbiological Methods*, 27(1):49–61.

Kotula, J. W., Kerns, S. J., Shaket, L. A., Siraj, L., Collins, J. J., Way, J. C., and Silver, P. A. (2014). Programmable bacteria detect and record an environmental signal in the mammalian gut. *Proc. Nat. Acad. Sci. U.S.A.*, 111(13):4838–43.

Kramer, B. P. and Fussenegger, M. (2005). Hysteresis in a synthetic mammalian gene network. *Proc. Nat. Acad. Sci. U.S.A.*, 102(27):9517–9522.

Kramer, B. P., Viretta, A. U., Daoud-El-Baba, M., Aubel, D., Weber, W., and Fussenegger, M. (2004). An engineered epigenetic transgene switch in mammalian cells. *Nat. Biotechnol.*, 22(7):867–870.

Kubitschek, H. E. (1990). Cell volume increase in escherichia coli after shifts to richer media. *J. Bacteriol.*, 172(1):94–101.

Kudla, G., Murray, A. W., Tollervey, D., and Plotkin, J. B. (2009). Coding-sequence determinants of gene expression in *Escherichia coli*. *Science*, 324(5924):255–8.

Kung, Y., Runguphan, W., and Keasling, J. D. (2012). From fields to fuels: recent advances in the microbial production of biofuels. *ACS Synth. Biol.*, 1(11):498–513.

Lahav, G., Rosenfeld, N., Sigal, A., Geva-Zatorsky, N., Levine, A. J., Elowitz, M. B., and Alon, U. (2004). Dynamics of the p53-mdm2 feedback loop in individual cells. *Nat. Genet.*, 36(2):147–150.

Lamont, I., Brumby, A. M., and Egan, J. B. (1989). UV induction of coliphage 186: Prophage induction as an SOS function. *Proc. Nat. Acad. Sci. U.S.A.*, 86(14):5492–5496.

Lamont, I., Richardson, H., Carter, D. R., and Egan, J. B. (1993). Genes for the establishment and maintenance of lysogeny by the temperate coliphage 186. *J. Bacteriol.*, 175(16):5286–5288.

Lampariello, F. and Aiello, A. (1998). Complete mathematical modeling method for the analysis of immunofluorescence distributions composed of negative and weakly positive cells. *Cytometry*, 32(3):241–54.

Le Meur, N. (2013). Computational methods for evaluation of cell-based data assessment–Bioconductor. *Curr. Opin. Biotechnol.*, 24(1):105–11.

Lee, T. I., Rinaldi, N. J., Robert, F., Odom, D. T., Bar-Joseph, Z., Gerber, G. K., Hannett, N. M., Harbison, C. T., Thompson, C. M., Simon, I., Zeitlinger, J., Jennings, E. G., Murray, H. L., Gordon, D. B., Ren, B., Wyrick, J. J., Tagne, J.-B., Volkert, T. L., Fraenkel, E., Gifford, D. K., and Young, R. A. (2002). Transcriptional regulatory networks in saccharomyces cerevisiae. *Science*, 298(5594):799–804.

Li, D. and Li, C. (2008). Noise-induced dynamics in the mixed-feedback-loop network motif. *Phys. Rev. E*, 77(1):011903.

Liang, S., Bipatnath, M., Xu, Y., Chen, S., Dennis, P., Ehrenberg, M., and Bremer, H. (1999). Activities of constitutive promoters in Escherichia coli. *J. Mol. Biol.*, 292(1):19–37.

Lim, H. N. and van Oudenaarden, A. (2007). A multistep epigenetic switch enables the stable inheritance of DNA methylation states. *Nat. Genet.*, 39(2):269–275.

Lin, T. I., Lee, J. C., and Hsieh, W. J. (2007). Robust mixture modeling using the skew t distribution. *Statistics and Computing*, 17(2):81–92.

Linn, T. and St Pierre, R. (1990). Improved vector system for constructing transcriptional fusions that ensures independent translation of *lacZ*. *J. Bacteriol.*, 172(2):1077–84.

Little, J. W. (1984). Autodigestion of lexa and phage lambda repressors. *Proc. Nat. Acad. Sci. U.S.A.*, 81(5):1375–9.

Little, J. W. and Mount, D. W. (1982). The SOS regulatory system of *Escherichia coli*. *Cell*, 29(1):11–22.

Lo, K., Hahne, F., Brinkman, R. R., and Gottardo, R. (2009). FlowClust: a Bioconductor package for automated gating of flow cytometry data. *BMC Bioinformatics*, 10:145.

Loinger, A. and Biham, O. (2009). Analysis of genetic toggle switch systems encoded on plasmids. *Phys. Rev. Lett.*, 103(6):068104.

Losick, R. and Desplan, C. (2008). Stochasticity and Cell Fate. *Science*, 320(5872):65–68.

Lu, P., Vogel, C., Wang, R., Yao, X., and Marcotte, E. M. (2007). Absolute protein expression profiling estimates the relative contributions of transcriptional and translational regulation. *Nat. Biotechnol.*, 25(1):117–24.

Lu, T. K. and Collins, J. J. (2007). Dispersing biofilms with engineered enzymatic bacteriophage. *Proc. Nat. Acad. Sci. U.S.A.*, 104(27):11197–202.

Lu, T. K. and Collins, J. J. (2009). Engineered bacteriophage targeting gene networks as adjuvants for antibiotic therapy. *Proc. Nat. Acad. Sci. U.S.A.*, 106(12):4629–34.

Lu, T. K., Khalil, A. S., and Collins, J. J. (2009). Next-generation synthetic gene networks. *Nat. Biotechnol.*, 27(12):1139–50.

Lutz, R. and Bujard, H. (1997). Independent and tight regulation of transcriptional units in *Escherichia coli* via the LacR/O, the TetR/O and AraC/I_1-I_2 regulatory elements. *Nucleic Acids Res.*, 25(6):1203–1210.

Maamar, H., Raj, A., and Dubnau, D. (2007). Noise in gene expression determines cell fate in Bacillus subtilis. *Science*, 317(5837):526–9.

Manen, D., Xia, G., and Caro, L. (1994). A locus involved in the regulation of replication in plasmid pSC101. *Mol. Microbiol.*, 11(5):875–84.

Massa, P. E., Paniccia, A., Monegal, A., de Marco, A., and Rescigno, M. (2013). Salmonella engineered to express CD20-targeting antibodies and a drug-converting enzyme can eradicate human lymphomas. *Blood*, 122(5):705–14.

Maurer, R., Meyer, B., and Ptashne, M. (1980). Gene regulation at the right operator (or) bacteriophage lambda. i. or3 and autogenous negative control by repressor. *J Mol Biol*, 139(2):147–61.

McAdams, H. H. and Arkin, A. (1997). Stochastic mechanisms in gene expression. *Proc. Nat. Acad. Sci. U.S.A.*, 94(3):814–9.

McGinness, K. E., Baker, T. A., and Sauer, R. T. (2006). Engineering controllable protein degradation. *Mol. Cell.*, 22(5):701–7.

Mettetal, J. T., Muzzey, D., Pedraza, J. M., Ozbudak, E. M., and van Oudenaarden, A. (2006). Predicting stochastic gene expression dynamics in single cells. *Proc. Nat. Acad. Sci. U.S.A.*, 103(19):7304–9.

Mikeladze-Dvali, T., Wernet, M. F., Pistillo, D., Mazzoni, E. O., Teleman, A. A., Chen, Y.-W., Cohen, S., and Desplan, C. (2005). The growth regulators warts/lats and melted interact in a bistable loop to specify opposite fates in Drosophila R8 photoreceptors. *Cell*, 122(5):775–87.

Mileyko, Y., Joh, R. I., and Weitz, J. S. (2008). Small-scale copy number variation and large-scale changes in gene expression. *Proc. Nat. Acad. Sci. U.S.A.*, 105(43):16659–16664.

Miller, J. et al. (1972). *Experiments in molecular genetics*. Cold Spring Harbor Laboratory, Cold Spring Harbor, NY.

Milo, R., Shen-Orr, S., Itzkovitz, S., Kashtan, N., Chklovskii, D., and Alon, U. (2002). Network motifs: simple building blocks of complex networks. *Science*, 298(5594):824–827.

Mishra, D., Rivera, P. M., Lin, A., Del Vecchio, D., and Weiss, R. (2014). A load driver device for engineering modularity in biological networks. *Nat. Biotechnol.*, 32(12):1268–75.

Mitra, K., Carvunis, A.-R., Ramesh, S. K., and Ideker, T. (2013). Integrative approaches for finding modular structure in biological networks. *Nat. Rev. Genet.*, 14(10):719–32.

Moon, T. S., Lou, C., Tamsir, A., Stanton, B. C., and Voigt, C. A. (2012). Genetic programs constructed from layered logic gates in single cells. *Nature*, 491(7423):249–53.

Mukherji, S. and van Oudenaarden, A. (2009). Synthetic biology: understanding biological design from synthetic circuits. *Nat. Rev. Genet.*, 10(12):859–71.

Munsky, B., Neuert, G., and van Oudenaarden, A. (2012). Using gene expression noise to understand gene regulation. *Science*, 336(6078):183–7.

Mutalik, V. K., Guimaraes, J. C., Cambray, G., Lam, C., Christoffersen, M. J., Mai, Q.-A., Tran, A. B., Paull, M., Keasling, J. D., Arkin, A. P., and Endy, D. (2013). Precise and reliable gene expression via standard transcription and translation initiation elements. *Nat. Methods.*, 10(4):354–60.

Nakanishi, H., Pedersen, M., Alsing, A. K., and Sneppen, K. (2009). Modeling of the genetic switch of bacteriophage tp901-1: A heteromer of ci and mor ensures robust bistability. *J. Mol. Biol.*, 394(1):15–28.

Nandagopal, N. and Elowitz, M. B. (2011). Synthetic biology: integrated gene circuits. *Science*, 333(6047):1244–8.

Neuert, G., Munsky, B., Tan, R. Z., Teytelman, L., Khammash, M., and van Oudenaarden, A. (2013). Systematic identification of signal-activated stochastic gene regulation. *Science*, 339(6119):584–7.

Neufing, P. J., Shearwin, K. E., and Egan, J. B. (2001). Establishing lysogenic transcription in the temperate coliphage 186. *J. Bacteriol.*, 183(7):2376–2379.

Oehler, S., Eismann, E. R., Krämer, H., and Müller-Hill, B. (1990). The three operators of the *lac* operon cooperate in repression. *EMBO J.*, 9(4):973–9.

Oltvai, Z. N. and Barabási, A.-L. (2002). Systems biology. life's complexity pyramid. *Science*, 298(5594):763–4.

Oppenheim, A. B., Kobiler, O., Stavans, J., Court, D. L., and Adhya, S. (2005). Switches in bacteriophage lambda development. *Annu. Rev. Genet.*, 39:409–29.

Ozbudak, E., Thattai, M., Lim, H., Shraiman, B., and van Oudenaarden, A. (2004). Multistability in the lactose utilization network of Escherichia coli. *Nature*, 427(6976):737–740.

Ozbudak, E. M., Thattai, M., Kurtser, I., Grossman, A. D., and van Oudenaarden, A. (2002). Regulation of noise in the expression of a single gene. *Nat. Genet.*, 31(1):69–73.

Paddon, C. J., Westfall, P. J., Pitera, D. J., Benjamin, K., Fisher, K., McPhee, D., Leavell, M. D., Tai, A., Main, A., Eng, D., Polichuk, D. R., Teoh, K. H., Reed, D. W., Treynor, T., Lenihan, J., Fleck, M., Bajad, S., Dang, G., Dengrove, D., Diola, D., Dorin, G., Ellens, K. W., Fickes, S., Galazzo, J., Gaucher, S. P., Geistlinger, T., Henry, R., Hepp, M., Horning, T., Iqbal, T., Jiang, H., Kizer, L., Lieu, B., Melis, D., Moss, N., Regentin, R., Secrest, S., Tsuruta, H., Vazquez, R., Westblade, L. F., Xu, L., Yu, M., Zhang, Y., Zhao, L., Lievense, J., Covello, P. S., Keasling, J. D., Reiling, K. K., Renninger, N. S., and Newman, J. D. (2013). High-level semi-synthetic production of the potent antimalarial artemisinin. *Nature*, 496(7446):528–32.

Palani, S. and Sarkar, C. A. (2011). Synthetic conversion of a graded receptor signal into a tunable, reversible switch. *Mol. Syst. Biol.*, 7:480.

Palmer, A. C., Ahlgren-Berg, A., Egan, J. B., Dodd, I. B., and Shearwin, K. E. (2009). Potent transcriptional interference by pausing of RNA polymerases over a downstream promoter. *Mol. Cell*, 34(5):545–55.

Parks, D. R., Roederer, M., and Moore, W. A. (2006). A new "Logicle" display method avoids deceptive effects of logarithmic scaling for low signals and compensated data. *Cytometry A*, 69(6):541–51.

Patterson, G. H., Knobel, S. M., Sharif, W. D., Kain, S. R., and Piston, D. W. (1997). Use of the green fluorescent protein and its mutants in quantitative fluorescence microscopy. *Biophys. J.*, 73(5):2782–90.

Paulsson, J. (2005). Models of stochastic gene expression. *Physics of life reviews*, 2(2):157–175.

Paulsson, J. and Ehrenberg, M. (2000). Random signal fluctuations can reduce random fluctuations in regulated components of chemical regulatory networks. *Phys. Rev. Lett.*, 84(23):5447–50.

Paulsson, J. and Ehrenberg, M. (2001). Noise in a minimal regulatory network: plasmid copy number control. *Q Rev Biophys*, 34(1):1–59.

Pédelacq, J.-D., Cabantous, S., Tran, T., Terwilliger, T. C., and Waldo, G. S. (2006). Engineering and characterization of a superfolder green fluorescent protein. *Nat. Biotechnol.*, 24(1):79–88.

Petzold, L. (1983). Automatic selection of methods for solving stiff and nonstiff systems of ordinary differential equations. *SIAM journal on scientific and statistical computing*, 4(1):136–148.

Phillips, R., Kondev, J., Theriot, J., Orme, N., and Garcia, H. (2009). *Physical biology of the cell*. Garland Science New York.

Pinkett, H. W., Shearwin, K. E., Stayrook, S., Dodd, I. B., Burr, T., Hochschild, A., Egan, J. B., and Lewis, M. (2006). The structural basis of cooperative regulation at an alternate genetic switch. *Molecular Cell*, 21(5):605–615.

Pocock, M. (2007). Synthetic biology: Construction of a mixed feedback loop from components of bacteriophage 186. Honours thesis, University of Adelaide.

Prates, M. O., Lachos, V. H., and Cabral, C. R. B. (2013). mixsmsn: Fitting Finite Mixture of Scale Mixture of Skew-Normal Distributions. *Journal of Statistical Software*, 54(12):1–20.

Priest, D. G., Cui, L., Kumar, S., Dunlap, D. D., Dodd, I. B., and Shearwin, K. E. (2014). Quantitation of the DNA tethering effect in long-range DNA looping *in vivo* and *in vitro* using the Lac and λrepressors. *Proc. Nat. Acad. Sci. U.S.A.*, 111(1):349–54.

Prindle, A., Samayoa, P., Razinkov, I., Danino, T., Tsimring, L. S., and Hasty, J. (2012). A sensing array of radically coupled genetic 'biopixels'. *Nature*, 481(7379):39–44.

Prindle, A., Selimkhanov, J., Li, H., Razinkov, I., Tsimring, L. S., and Hasty, J. (2014). Rapid and tunable post-translational coupling of genetic circuits. *Nature*, 508(7496):387–91.

Ptashne, M. (2004). *A Genetic Switch: Phage Lambda Revisited*. CSHL Press.

Pyne, S., Hu, X., Wang, K., Rossin, E., Lin, T.-I., Maier, L. M., Baecher-Allan, C., McLachlan, G. J., Tamayo, P., Hafler, D. A., De Jager, P. L., and Mesirov, J. P. (2009). Automated high-dimensional flow cytometric data analysis. *Proc. Nat. Acad. Sci. U.S.A.*, 106(21):8519–24.

Qian, H. (2012). Cooperativity in cellular biochemical processes: noise-enhanced sensitivity, fluctuating enzyme, bistability with nonlinear feedback, and other mechanisms for sigmoidal responses. *Annu. Rev. Biophys.*, 41:179–204.

Qian, H. and Bishop, L. M. (2010). The chemical master equation approach to nonequilibrium steady-state of open biochemical systems: linear single-molecule enzyme kinetics and nonlinear biochemical reaction networks. *Int. J. Mol. Sci.*, 11(9):3472–500.

Qiu, P., Simonds, E. F., Bendall, S. C., Gibbs, Jr, K. D., Bruggner, R. V., Linderman, M. D., Sachs, K., Nolan, G. P., and Plevritis, S. K. (2011). Extracting a cellular hierarchy from high-dimensional cytometry data with SPADE. *Nat. Biotechnol.*, 29(10):886–91.

R Development Core Team (2012). *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria. ISBN 3-900051-07-0, `http://www.R-project.org/`.

Raj, A. and van Oudenaarden, A. (2008). Nature, nurture, or chance: stochastic gene expression and its consequences. *Cell*, 135(2):216–26.

Rao, C. V. and Arkin, A. P. (2003). Stochastic chemical kinetics and the quasi-steady-state assumption: application to the Gillespie algorithm. *J. Chem. Phys.*, 118(11):4999–5010.

Rao, C. V., Wolf, D. M., and Arkin, A. P. (2002). Control, exploitation and tolerance of intracellular noise. *Nature*, 420(6912):231–7.

Rasband, W. (1997–2011). ImageJ. U. S. National Institutes of Health, Bethesda, Maryland, USA. `http://imagej.nih.gov/ij/`.

Ray, J. C. J., Tabor, J. J., and Igoshin, O. A. (2011). Non-transcriptional regulatory processes shape transcriptional network dynamics. *Nat. Rev. Microbiol.*, 9(11):817–28.

Rice, J. A. (2007). *Mathematical statistics and data analysis*. Duxbury Advanced Series. Thomson, Belmont, CA, 3rd edition.

Ritz, C. and Streibig, J. C. (2005). Bioassay Analysis using R. *Journal of Statistical Software*, 12(5). `http://www.bioassay.dk`.

Ro, D.-K., Paradise, E. M., Ouellet, M., Fisher, K. J., Newman, K. L., Ndungu, J. M., Ho, K. A., Eachus, R. A., Ham, T. S., Kirby, J., Chang, M. C. Y., Withers, S. T., Shiba, Y., Sarpong, R., and Keasling, J. D. (2006). Production of the antimalarial drug precursor artemisinic acid in engineered yeast. *Nature*, 440(7086):940–3.

Robinson, J. P., Rajwa, B., Patsekin, V., and Davisson, V. J. (2012). Computational analysis of high-throughput flow cytometry data. *Expert Opin. Drug Discov.*, 7(8):679–93.

Ronen, M., Rosenberg, R., Shraiman, B. I., and Alon, U. (2002). Assigning numbers to the arrows: parameterizing a gene regulation network by using accurate expression kinetics. *Proc. Nat. Acad. Sci. U.S.A.*, 99(16):10555–10560.

Rosario, K. and Breitbart, M. (2011). Exploring the viral world through metagenomics. *Curr. Opin. Virol.*, 1(4):289–97.

Rosenfeld, N., Elowitz, M. B., and Alon, U. (2002). Negative autoregulation speeds the response times of transcription networks. *J. Mol. Biol.*, 323(5):785–93.

Rosenfeld, N., Young, J. W., Alon, U., Swain, P. S., and Elowitz, M. B. (2007). Accurate prediction of gene feedback circuit behavior from component properties. *Mol. Syst. Biol.*, 3:143.

Ruder, W. C., Lu, T., and Collins, J. J. (2011). Synthetic biology moving into the clinic. *Science*, 333(6047):1248–52.

Salis, H. and Kaznessis, Y. (2005). Accurate hybrid stochastic simulation of a system of coupled chemical or biochemical reactions. *J. Chem. Phys.*, 122(5):54103.

Sezonov, G., Joseleau-Petit, D., and D'Ari, R. (2007). *Escherichia coli* physiology in Luria-Bertani broth. *J. Bacteriol.*, 189(23):8746–9.

Shahrezaei, V. and Swain, P. S. (2008). Analytical distributions for stochastic gene expression. *Proc. Nat. Acad. Sci. U.S.A.*, 105(45):17256–61.

Shaner, N. C., Steinbach, P. A., and Tsien, R. Y. (2005). A guide to choosing fluorescent proteins. *Nat. Methods*, 2(12):905–9.

Shea, M. A. and Ackers, G. K. (1985). The O_R control system of bacteriophage lambda: A physical-chemical model for gene regulation. *J. Mol. Biol.*, 181(2):211–30.

Shearwin, K. E., Brumby, A. M., and Egan, J. B. (1998). The Tum protein of coliphage 186 is an antirepressor. *J. Biol. Chem.*, 273(10):5708–5715.

Shearwin, K. E. and Egan, J. B. (1996). Purification and self-association equilibria of the lysis-lysogeny switch proteins of coliphage 186. *J. Biol. Chem.*, 271(19):11525–11531.

Shen-Orr, S. S., Milo, R., Mangan, S., and Alon, U. (2002). Network motifs in the transcriptional regulation network of escherichia coli. *Nat. Genet.*, 31(1):64–68.

Shetty, R. P., Endy, D., and Knight, Jr, T. F. (2008). Engineering BioBrick vectors from BioBrick parts. *J. Biol. Eng.*, 2:5.

Shi, J. and Biek, D. P. (1995). A versatile low-copy-number cloning vector derived from plasmid F. *Gene*, 164(1):55–58.

Shimoni, Y., Friedlander, G., Hetzroni, G., Niv, G., Altuvia, S., Biham, O., and Margalit, H. (2007). Regulation of gene expression by small non-coding rnas: a quantitative view. *Mol. Syst. Biol.*, 3:138.

Shotland, Y., Koby, S., Teff, D., Mansur, N., Oren, D. A., Tatematsu, K., Tomoyasu, T., Kessel, M., Bukau, B., Ogura, T., and Oppenheim, A. B. (1997). Proteolysis of the phage λ CII regulatory protein by FtsH (HflB) of *Escherichia coli*. *Mol. Microbiol.*, 24(6):1303–10.

Siegal-Gaskins, D., Mejia-Guerra, M. K., Smith, G. D., and Grotewold, E. (2011). Emergence of switch-like behavior in a large family of simple biochemical networks. *PLoS Comput. Biol.*, 7(5):e1002039.

Simons, R. W., Houman, F., and Kleckner, N. (1987). Improved single and multicopy lac-based cloning vectors for protein and operon fusions. *Gene*, 53(1):85–96.

Smidtas, S., Schachter, V., and Kepes, F. (2006). The adaptive filter of the yeast galactose pathway. *J. Theor. Biol.*, 242(2):372–381.

Smolen, P., Baxter, D. A., and Byrne, J. H. (2000). Mathematical modeling of gene networks. *Neuron*, 26(3):567–80.

Sneppen, K., Dodd, I. B., Shearwin, K. E., Palmer, A. C., Schubert, R. A., Callen, B. P., and Egan, J. B. (2005). A mathematical model for transcriptional interference by rna polymerase traffic in escherichia coli. *J. Mol. Biol.*, 346(2):399–409.

Sneppen, K., Krishna, S., and Semsey, S. (2010). Simplified models of biological networks. *Annu. Rev. Biophys.*, 39:43–59.

Sneppen, K. and Zocchi, G. (2005). *Physics in Molecular Biology*. Cambridge University Press.

Soetaert, K. and Herman, P. M. (2009). *A Practical Guide to Ecological Modelling: Using R as a Simulation Platform*. Springer.

Soetaert, K. and Petzoldt, T. (2010). Inverse modelling, sensitivity and monte carlo analysis in R using package FME. *Journal of Statistical Software*, 33(3):1–28.

Soetaert, K., Petzoldt, T., and Setzer, R. (2010). Solving differential equations in R: Package deSolve. *Journal of Statistical Software*, 33(9):1–25.

Sprinzak, D. and Elowitz, M. B. (2005). Reconstruction of genetic circuits. *Nature*, 438(7067):443–448.

St-Pierre, F., Cui, L., Priest, D. G., Endy, D., Dodd, I. B., and Shearwin, K. E. (2013). One-step cloning and chromosomal integration of DNA. *ACS Synth. Biol.*, 2(9):537–41.

Stricker, J., Cookson, S., Bennett, M., Mather, W., Tsimring, L., and Hasty, J. (2008). A fast, robust and tunable synthetic gene oscillator. *Nature*, 456:516–519.

Stueber, D. and Bujard, H. (1982). Transcription from efficient promoters can interfere with plasmid replication and diminish expression of plasmid specified genes. *EMBO J.*, 1(11):1399–404.

Süel, G. M., Garcia-Ojalvo, J., Liberman, L. M., and Elowitz, M. B. (2006). An excitable gene regulatory circuit induces transient cellular differentiation. *Nature*, 440(7083):545–550.

Swain, P. S., Elowitz, M. B., and Siggia, E. D. (2002). Intrinsic and extrinsic contributions to stochasticity in gene expression. *Proc. Nat. Acad. Sci. U.S.A.*, 99(20):12795–800.

Tan, C., Marguet, P., and You, L. (2009). Emergent bistability by a growth-modulating positive feedback circuit. *Nat. Chem. Biol.*, 5(11):842–8.

Taniguchi, Y., Choi, P. J., Li, G.-W., Chen, H., Babu, M., Hearn, J., Emili, A., and Xie, X. S. (2010). Quantifying *E. coli* proteome and transcriptome with single-molecule sensitivity in single cells. *Science*, 329(5991):533–8.

Terejanu, G., Singla, P., Singh, T., and Scott, P. D. (2008). Uncertainty propagation for nonlinear dynamic systems using Gaussian mixture models. *Journal of Guidance, Control, and Dynamics*, 31(6):1623–1633.

Thattai, M. and van Oudenaarden, A. (2001). Intrinsic noise in gene regulatory networks. *Proc. Nat. Acad. Sci. U.S.A.*, 98(15):8614–8619.

Thieffry, D., Huerta, A. M., Pérez-Rueda, E., and Collado-Vides, J. (1998). From specific gene regulation to genomic networks: a global analysis of transcriptional regulation in *Escherichia coli*. *Bioessays*, 20(5):433–40.

Tigges, M., Marquez-Lago, T. T., Stelling, J., and Fussenegger, M. (2009). A tunable synthetic mammalian oscillator. *Nature*, 457(7227):309–312.

To, T.-L. and Maheshri, N. (2010). Noise can induce bimodality in positive transcriptional feedback loops without bistability. *Science*, 327(5969):1142–5.

Trusina, A., Sneppen, K., Dodd, I. B., Shearwin, K. E., and Egan, J. B. (2005). Functional alignment of regulatory networks: a study of temperate phages. *PLoS Comput. Biol.*, 1(7):e74.

Tsuru, S., Yasuda, N., Murakami, Y., Ushioda, J., Kashiwagi, A., Suzuki, S., Mori, K., Ying, B.-W., and Yomo, T. (2011). Adaptation by stochastic switching of a monostable genetic circuit in escherichia coli. *Mol. Syst. Biol.*, 7:493.

van Kampen, N. G. (2007). *Stochastic Processes in Physics and Chemistry*. North Holland, 3rd edition.

Venturelli, O. S., El-Samad, H., and Murray, R. M. (2012). Synergistic dual positive feedback loops established by molecular sequestration generate robust bimodal response. *Proc. Nat. Acad. Sci. U.S.A.*, 109(48):E3324–33.

Voigt, C. A. (2006). Genetic parts to program bacteria. *Curr. Opin. Biotechnol.*, 17(5):548–557.

Waldo, G. S., Standish, B. M., Berendzen, J., and Terwilliger, T. C. (1999). Rapid protein-folding assay using green fluorescent protein. *Nat. Biotechnol.*, 17(7):691–5.

Wall, M. E., Hlavacek, W. S., and Savageau, M. A. (2003). Design principles for regulator gene expression in a repressible gene circuit. *J. Mol. Biol.*, 332(4):861–76.

Wall, M. E., Hlavacek, W. S., and Savageau, M. A. (2004). Design of gene circuits: lessons from bacteria. *Nat. Rev. Genet.*, 5(1):34–42.

Wang, H., Dodd, I. B., Dunlap, D. D., Shearwin, K. E., and Finzi, L. (2013). Single molecule analysis of dna wrapping and looping by a circular 14mer wheel of the bacteriophage 186 ci repressor. *Nucleic Acids Res.*, 41(11):5746–56.

Way, J. C., Collins, J. J., Keasling, J. D., and Silver, P. A. (2014). Integrating biological redesign: where synthetic biology came from and where it needs to go. *Cell*, 157(1):151–61.

Wilkinson, D. J. (2006). *Stochastic Modelling for Systems Biology*. Chapman & Hall/CRC.

Wilkinson, D. J. (2009). Stochastic modelling for quantitative description of heterogeneous biological systems. *Nat. Rev. Genet.*, 10(2):122–33.

Woods, W. H. and Egan, J. B. (1974). Prophage induction of noninducible coliphage 186. *J. Virol.*, 14(6):1349–1356.

Xia, G. X., Manen, D., Goebel, T., Linder, P., Churchward, G., and Caro, L. (1991). A copy-number mutant of plasmid pSC101. *Mol. Microbiol.*, 5(3):631–40.

Yeger-Lotem, E. and Margalit, H. (2003). Detection of regulatory circuits by integrating the cellular networks of protein-protein interactions and transcription regulation. *Nucleic Acids Res.*, 31(20):6053–6061.

Yeger-Lotem, E., Sattath, S., Kashtan, N., Itzkovitz, S., Milo, R., Pinter, R. Y., Alon, U., and Margalit, H. (2004). Network motifs in integrated cellular networks of transcription-regulation and protein-protein interaction. *Proc. Nat. Acad. Sci. U.S.A.*, 101(16):5934–5939.

Yeh, B. J. and Lim, W. A. (2007). Synthetic biology: lessons from the history of synthetic organic chemistry. *Nat. Chem. Biol.*, 3(9):521–525.

Yu, J., Xiao, J., Ren, X., Lao, K., and Xie, X. S. (2006). Probing gene expression in live cells, one protein molecule at a time. *Science*, 311(5767):1600–1603.

Zeng, L., Skinner, S. O., Zong, C., Sippy, J., Feiss, M., and Golding, I. (2010). Decision making at a subcellular level determines the outcome of bacteriophage infection. *Cell*, 141(4):682–91.

Zong, C., So, L.-h., Sepúlveda, L. A., Skinner, S. O., and Golding, I. (2010). Lysogen stability is determined by the frequency of activity bursts from the fate-determining gene. *Mol. Syst. Biol.*, 6:440.

Zwietering, M. H., Jongenburger, I., Rombouts, F. M., and van 't Riet, K. (1990). Modeling of the bacterial growth curve. *Appl. Environ. Microbiol.*, 56(6):1875–81.