A Systematic Investigation of Multi-Objective Evolutionary Algorithms Applied to the Water Distribution System Problem

Peng Wang

B.Eng., M.Eng.

Thesis submitted in fulfilment of the requirements for the degree of Doctor of Philosophy

The University of Adelaide

Faculty of Engineering, Computer and Mathematical Sciences

School of Civil, Environmental and Mining Engineering

Abstract

Water distribution systems (WDSs) are one of society's most important infrastructure assets. They consist of a great number of pumps, valves, junctions and a tremendous number of pipes that connect these nodes within the system, all of which induce a significant capital cost at the time of construction. However, there is no singular option for designing a WDS, and each potential design affects the cost and performance of the system differently (i.e., the pressure at each node and flow rates for each pipe). To identify solutions with a better trade-off between the cost and performance, multi-objective evolutionary algorithms (MOEAs) provide a robust optimisation tool to solve this type of problem. This PhD thesis focuses on improving and developing a more effective MOEA for WDS problems, and optimisation problems in general. The first stage of the research is to study the impact of select critical processes in MOEAs on algorithm performance and understand the reasons behind the performance observations. There are two chapters related to the first stage. The second stage is to develop a proposed General Multi-Objective Evolutionary Algorithm (GMOEA) and compare this with existing MOEAs for WDS problems. This is associated with the third content chapter.

In the first paper, the impact of the operators on an algorithm's performance has been studied. The operators are the key component for exchange of information between solutions in populations to produce offspring solutions, thereby exploring alternative regions of the search space. These have a significant impact on an algorithm's search behaviour. However, the composition and number of operators that should be included in an MOEA is generally fixed, based on choices made by the developers of these algorithms. To explore this issue, an assessment was conducted via comprehensive numerical experiments that isolate the influence of the size of the operator set, as well as its composition. In addition, the relative influence of other search processes affecting search behaviour (e.g., the selection strategy and hyperheuristic) have been studied. It has been found that operator set size is a dominant factor affecting algorithm performance, having a greater influence than operator set composition and other search processes affecting algorithm search behaviour. Moreover, it was also found that an existing MOEAs' performance can be improved by simply increasing the number of operators used within the algorithm. This finding can be applied to justify the usage of operators for designing a new MOEA in the future.

In the second paper, a new convex hull contribution selection strategy for population-based MOEAs (termed CHC_{Gen}) has been proposed and compared with existing MOEAs in order to study the impact of the selection strategy on MOEA performance. It has been found that the CHC_{Gen} selection strategy is able to emphasise selection of the population of solutions on the convex hull of the non-dominated set of solutions. The CHC_{Gen} selection strategy has demonstrated that it can also improve an existing MOEAs' performance. The finding suggests different selection strategies have an impact on MOEA performance. In addition, CHC_{Gen} can be used for developing a new MOEA in the future.

In the third paper, a new multi-objective evolutionary algorithm, called GMOEA(CHC_{Gen},12,T,A)[1] has been proposed by conducting comprehensive numerical experiments to determine the optimised component configuration for each MOEA process. The components considered within the algorithm construction include: the selection strategy, hyperheuristic, and operator set size. The numerical experiments not only explore the impact of each process's

[1] Refer to Chapter 4 for the definition of the notation

component on algorithm performance comprehensively, but also investigate the correlation of each pairwise combination of the process's components. In addition, the optimal form of the algorithm GMOEA(CHC_{Gen},12,T,A) was compared with seven other existing MOEAs with an extended computational budget for a range of WDS problems. From the results, GMOEA(CHC_{Gen},12,T,A) was shown not only to have outperformed all other MOEAs considered, but also to find a greater number of new Pareto front solutions for intermediate and large scale problems.

Statement of Originality

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 acknowledge 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.

I acknowledge the support I have received for my research through the provision of an Australian Government Research Training Program Scholarship.

Signed: ………… ……………Date: 20/11/2020

This page is intentionally blank.

Acknowledgements

I would like to express my gratitude and thanks to my supervisors Dr Aaron C. Zecchin and Prof. Holger R. Maier. Thanks go to Aaron for his creative mind for proposing research questions and his great support with the mathematical problems that emerged during my PhD program. I gained a deep understanding about how to propose a research question and how to solve them. Thanks go to Holger for his substantial professional suggestions, solid logic and innovative way of thinking. I learned the effective way to conduct critical thinking. Thanks to my supervisors for providing substantial comments and feedback for the manuscripts throughout my PhD journey, which improved my writing skills significantly.

I would like to thank Prof. Feifei Zheng for kindly providing me with an opportunity and funding for visiting Zhejiang University for one month and for his contributions to my papers.

I would like to thank the postgraduate students of the School for their friendship. I thank the staff in the School for their support to make my PhD program organised. I thank Dr Alison-Jane Hunter for her consistent help to improve my grammar skills during the candidature. I thank my best friend Dr Wei Zeng for his friendship and help to develop my career plan during the candidature.

I would like to thank my parents and relatives, for their love and support. I also express my deep gratitude to my wife Xinyi Mao, who has been always being patient and supportive. My life is fantastic because of all of you.

This page is intentionally blank.

Table of Contents

Publications Arising from This Thesis

The following peer-reviewed journal paper and unpublished manuscripts are the major outcomes of this research and they form the main body of this thesis.

1. Wang, P., Zecchin, A. C., Maier, H. R., Zheng, F., & Newman, J. P. (2020). Do existing multiobjective evolutionary algorithms use a sufficient number of operators? An empirical investigation for water distribution design problems. Water Resources Research, 55, e2019WR026031. https://doi.org/10.1029/2019WR026031
2. Wang, P., Zecchin, A. C., Maier, H. R., (2020). A New Proposed Convex Hull Selection Strategy: Study on the Impact of Selection Strategy on Generational MOEA Performance for Water Distribution Design Problems. (Unpublished manuscript)
3. Wang, P., Zecchin, A. C., Maier, H. R., (2020). A Multi-Objective Evolutionary Algorithm Component Investigation for Solving Water Distribution System Optimisation Problems. (Unpublished manuscript)

List of Tables

This page is intentionally blank

List of Figures

Chapter 1 Introduction

1.1 Background

Multi-Objective Evolutionary Algorithms (MOEAs) have been applied to solve WDS problems for over two decades. Their effectiveness at solving this type of optimisation problem is because these algorithms have the ability to adjust the way they search through the solution spaces by either intensifying the search in promising regions (i.e., exploiting good solutions) or diversifying the search in less promising regions (e.g., exploring the solutions space more widely), enabling them to perform well on problems with different characteristics (Maier et al., 2019). Such search behaviour responses can be fine-tuned by closely considering and adjusting the components of each process of an MOEA (e.g. the mutation strategy component in the reproduction process, or the selection strategy component in the parental selection process). Hence, many MOEAs with advanced features have been proposed to improve an algorithm's performance. However, the framework of most MOEAs is generally consistent in terms of their processes; hence, the study of how each process affects MOEA performance is not yet sufficiently broad or generalised as authors typically only propose a particular algorithm (which is a set collection of process components) rather than study the broader question of what components or combinations of components are best. Moreover, it is hard to understand questions like why one algorithm outperforms another one without considering the impact that individual components have on performance. In this thesis, a General Multi-objective Evolutionary Algorithm (GMOEA) is used to isolate the impact of each MOEA's process on the search behaviour. The GMOEA provides a general MOEA framework within which components can be incorporated or replaced to enable the construction of a very broad array of customised algorithms. After understanding how each MOEA process affects the performance, a new algorithm GMOEA(CHC_{gen},12,T,A) was proposed and compared with the seven popular existing MOEAs to solve six WDS problems with different levels of complexity.

1.1.1 Significance of multi-objective evolutionary algorithms optimising water distribution system problems

Water distribution systems (WDSs) are designed to transport potable water from post-treatment water sources to consumers (Zecchin et al., 2005). As the cost of construction and maintenance of pipelines for the water supply is significant, there is an increasing desire to achieve a high level of effectiveness for each dollar spent (Simpson et al., 1994). In general, a WDS pipes' diameters are treated as the decision variables and inform the constraints that determine the feasibility of a design (i.e., provide adequate pressure head) (Zecchin et al., 2005).

In the past, WDS problems were treated as single-objective optimisation problems. The first objective is minimising the capital cost of the pipes while satisfying the network's constraint or second objective of minimum residual pressure head. However, the limitations of this formulation have been criticized broadly as being too simplistic for the design of real systems. The reason is that it is difficult to balance the weights for any objectives. If the first objective is weighted too heavily, the outcome optimized solution is less reliable (Engelhardt et al., 2000; Fu et al., 2012; Walski 2001). Specifically, by combining the two objectives into a single objective, the trade-off information in each objective is lost (Singh et al., 2003). Consequently, it was necessary to develop a multi-objective formulation for WDS problems. In order to achieve this, a great number of indicators have been proposed as a second objective. For example, minimising the total pressure deficit (Cheung et al., 2003); or minimising the number of nodes with a head deficiency (Farmani et al., 2004). However, these formulations are not necessarily compliant with looped network designs, which are reliable configurations under abnormal conditions (e.g., pipe burst). In an influential paper, Prasad & Park (2004) proposed an indicator called network resilience. Network resilience considers the effect of redundancy on a pipe network and maximizing this indicator can ensure reliable loops. In recent years, much work has been undertaken that used cost and network resilience as objective functions for WDS problems (Wang et al., 2015; Zheng et al., 2016; Wang et al., 2017; Jahanpour et al., 2018; Wang et al., 2020a).

Given that the search space of possible design solutions for a network system is very large, it is computationally infeasible to find the global optima by enumerating each possible design. For example, the New York tunnel network consists of 21 pipes with 16 diameter options for each pipe. The entire search space size is about 1.93×10^{25}. Thus, it would take 6.12×10^{16} years to evaluate all the solutions, given that one solution evaluation takes 0.1 seconds of clock time. In addition, the objectives and constraints are all nonlinear functions of the decision variables (Jahanpour et al., 2018). The NP-hard nature of this type of problem is a challenge to tackle, especially for large, real-world networks (Wang et al., 2015).

Considering the nature of the problem type, Simpson et al., (1994) firstly applied the genetic algorithm (GA) to WDS problems (i.e., the single objective optimization problem) and demonstrated the performance of the GA outweighed other deterministic optimisation methods, such as linear programming (Schaake & Lai, 1969), and nonlinear programming (Murtagh & Saunders 1987). Then, as mentioned earlier, as multi-objectives have been adopted in the WDS problem, Deb et al., (2002b) proposed the nondominated sorting genetic algorithm-II (NSGA-II), which has been applied to WDS and has shown effective performance (Jourdan et al., 2005; Khu & Keedwell, 2005). NSGA-II is an effective MOEA that has been widely used as a benchmark MOEA in water engineering (Farmani et al., 2004). Moreover, it serves as the prototype of some state-of-the-art MOEAs (excluding some unique features). Thereafter, many MOEAs were developed to achieve effective performance on various types of problems. As inspired from natural adaptive systems, the search behaviour should adapt to different problems. This is achieved by implementing multiple operators, given different operators have different characteristics, therefore having different search behaviours. In addition, the degree to which each of the operators contributes to the search at each iteration can be controlled with the aid of hyperheuristics, which are high-level automated search methodologies for selecting the most appropriate lower-level operators (or heuristics) (Burke et al., 2013; Drake et al., 2019). This type of MOEA is able to change the search behaviour to adapt to a problem's characteristics, thereby improving the algorithm's performance. For the sake of understanding MOEA processes, the general structure of the state-of-the-art MOEAs is outlined in Figure 1-1. As can be seen, at the beginning of the optimisation process, an initial set of solutions is randomly generated to form the population. There are three key processes undertaken within an iteration. They are (i) parent selection, (ii) reproduction, and (iii) survivor selection. Subject to the parent selection process, some solutions are selected from the population as parent solutions, which have the opportunity to reproduce and create offspring. In the reproduction process, the new offspring solutions are produced from the selected parent solutions by use of one or more operators (e.g., cross-over from the parents). When using multiple operators, the degree to which an operator contributes to the search at each generation can be controlled with the aid of a hyperheuristic. Thereafter, the new population in the current generation and the new offspring are collated to form a combined set. Then, replacement is carried out to select the successful solutions from the combined set to form the next generation's population. The above process is repeated until certain termination criteria are met, such as the execution of a fixed number of generations, or no better solutions being identified within a given time interval.

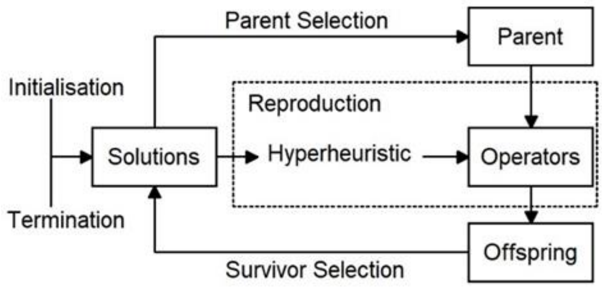


Figure 1-1. Generic multi-objective evolutionary optimisation process

Examples of such algorithms are AMALGAM (Vrugt et al., 2007), Borg (Hadka & Reed, 2013), and a genetically adaptive leaping algorithm for approximation and diversity, GALAXY (Wang et al., 2017). These algorithms inherit some features of NSGA-II, and implement multiple operators with a hyperheuristic to favour more successful operators automatically to carry out reproduction. The MOEAs mentioned above have shown successful and effective performance for WDS problems (Wang et al., 2015, 2017; Zheng et al., 2016; Jahanpour et al., 2018; Wang et al., 2020a).

1.1.2 MOEA Development

In the past, the development of an MOEA has been motivated by addressing the limitations of computational efficiency or to achieve certain targets to improve algorithm performance. For example, Deb et al., (2002b) proposed NSGA-II in order to (i) reduce the high computational complexity of nondominated sorting; (ii) achieve elitism preservation; and (iii) develop an effective diversity maintenance strategy. These objectives were fulfilled by (i) the fast-nondominated sorting approach; (ii) combination of offspring with the last generation's population to retain elite solutions; and (iiii) use of the crowding distance metric, respectively.

In other relevant work, Vrugt et al., (2007) found that the nature of fitness landscapes is considerably different in different problems. To improve MOEA performance, it was considered necessary to find a way to customise the search behaviour to cater for the different problem characteristics during the search. Given that MOEAs are able to be shown different search behaviours via the use of a range of different operators (Maier et al., 2014; Zecchin et al., 2012; Zheng et al., 2015, 2017a), it is possible to tune an algorithm's search behaviour by dynamically adjusting the degree to which different operators contribute to the identification of better solutions throughout the search. For many algorithms employing such approaches, this dynamic adjustment is controlled by a hyperheuristic (Burke et al., 2013), which tracks the historical performance of an operator and allocates computational resources to the operator based on this performance. Inspired by models of adaptation in natural systems, AMALGAM was proposed to use multiple operators, assisted by a hyperheuristic to tune each operator's utilisation rate (Vrugt et al., 2007) during the search. This innovative hyperheuristic allows the algorithm to adapt to the current search by utilising more successful operators and improving algorithm performance.

Hadka & Reed (2013) comprehensively studied the weaknesses of existing MOEAs for high objective dimension optimisation problems. Key issues include the: (i) lack of an appropriate non-dominance relationship for high dimension objectives; (ii) lack of an appropriate diversity maintenance strategy, (iii) risk of deterioration, in terms where the elite solutions are replaced by worse solutions during the search; and (iv) reduction of the work for parameterisation for the crossover rate and mutation rate of operators, for example. Borg was designed to address these issues, and is equipped with many new features such as an ε-non-dominance archive to store the elite solutions to avoid deterioration; multiple operators with a hyperheuristic to adapt the search to different problems' characteristics and reduce parameterisation by using the six operators with a hyperheuristic to adapt the search to different problems' characteristics, such as ε-progress and an adaptive population sizing operator (hyperheuristic) to improve the diversity of the search.

The above algorithms have shown effective performance not only on a wide range of applications and test functions (Asadzadeh & Tolson 2012; Hadka & Reed, 2012, 2013, 2015; Zeff et al., 2016; Zhang et al., 2010), but also on WDS problems (Wang et al., 2015; 2017; Zheng et al., 2016). Following on from the works undertaken, Wang et al., (2017) aimed to improve MOEA performance by optimising the components of an MOEA. This work represented the belief that the existing MOEA's structure is effective, but that tailoring the components would benefit algorithm performance. Thus, GALAXY was proposed, equipped with a new survivor selection strategy and search operators, which were tailored for WDS problems (i.e., using operators that only work in the discrete search space). The idea of the algorithm development is that different search processes have different search behaviours, thereby affecting an algorithm's performance. By conducting a numerical comparison study, it was demonstrated that GALAXY outperformed the aforementioned algorithms on WDS problems (Wang et al., 2017).

1.1.3 Limitations of existing MOEAs

Key remaining issues in the applied EA field is that it is difficult to ascertain the reason or justification for a particular option of each MOEA component, aside from the overall end-of-run metrics, which only indicate the collective influence of all components. For example, it is unclear how many operators are sufficient for improving an algorithm's performance. Also, it is unclear how different selection strategies affect search behaviour, and further, which hyperheuristics are able to improve algorithm performance effectively. These questions are the research gaps of this thesis and are introduced in detail in the following subsections.

1.1.3.1 The relative influence of size and composition of the operator set

Evolutionary algorithms are able to achieve the diverse searching behaviour outlined above via the use of a range of operators (Maier et al., 2014; Zecchin et al., 2012; Zheng et al., 2015, 2017a), which provide different search behaviours for searching through the solution space. For the existing MOEAs, how many and which operators are used in a particular algorithm is generally fixed, based on the choices made by the developers of these algorithms. For example, NSGA-II, which is one of the most common widely used EAs, uses two operators [simulated binary crossover (SBX) (Deb & Agrawal., 1994) and polynomial mutation (PM) (Deb & Agrawal., 1994)]; whereas the more recently-developed algorithms AMALGAM (Vrugt & Robinson, 2007), Borg (Hadka & Reed, 2013) and GALAXY (Wang et al., 2017) use six [SBX, PM, particle swarm optimisation (PSO) (Kennedy & Eberhart, 1995), turbulence factor (TF) (Pulido et al., 2004), differential evolution (DE) (Storn & Price, 1997) and adaptive metropolis strategy (AMS) (Haario et al., 2001)], seven [parent-centric crossover (PCX) (Deb et al., 2002a), simplex crossover (SPX) (Tsutsui et al., 1999), unimodal distribution crossover (UNDX) (Kita et al., 1999), uniform mutation (UM) (Michalewicz, 1992), SBX, PM and DE] and six [dither creeping (DC) (Wang et al., 2017), gaussian mutation (GM) (Wang et al., 2017), SBX, DE, TF and UM] operators, respectively. The issue of how many and which operators are included in an algorithm is likely to have a greater influence on algorithm performance than other search processes (as long as reasonable parameter values are selected for these processes) (Soria-Alcarez et al., 2017). Consequently, it is somewhat surprising that the influence of the different number of operators on the performance of EAs has received very limited attention, with only Vrugt et al., (2009) exploring this issue. However, Vrugt et al., (2009) only considered five candidate operators, making it possible to investigate the impact of all possible operator combinations of this limited set. This is not the case when considering a larger number of operators, such as those used in current state-of-the-art algorithms. The above shortcoming, with knowledge of the appropriate operator set, is discussed and addressed in Chapter 2.

1.1.3.2 The relative influence of the selection strategy

The selection strategy is a key component of an MOEA, which determines the composition of a population, and thereby the evolutionary search process overall, which imitates natural selection by granting fitter individuals an increasing opportunity to reproduce (Yu & Gen, 2010). The selection strategy is an important process in the evolution of the population, as it needs to be designed to drive convergence to increasingly fit regions of the search space (though elitism, for example), whilst avoiding pre-mature convergence to sub-optimal regions (through maintaining population diversity) (Back, 1996; Hanne, 1999). Over the past 20 years, many selection strategies have been proposed and shown to be effective in different MOEAs. For example, Emmerich et al., (2005) applied a hypervolume contribution (HVC) selection strategy (Knowles et al., 2003) to SMS-EMOEA. The results show it outperformed NSGA-II (which uses a crowding distance (CD) selection strategy (Deb et al., 2002b)); however, this thesis did not isolate the impact of the selection strategy from those of the other MOEA components, which poses a difficulty in attributing the performance difference to the proposed selection strategy. Consequently, investigation into the relative influence of a selection strategy on an algorithm's performance still needs to be addressed. A detailed review of the existing selection strategies is outlined in Chapter 3, where the aim is understanding how a selection strategy can affect search behaviour, thereby affecting the algorithm's performance. Moreover, Chapter 3 proposes a new convex hull contribution selection strategy for population-based MOEAs (termed CHC_{Gen}) and this is shown to be the best performing selection strategy component - performing better than the other existing selection strategies considered in this thesis.

1.1.3.3 The component combination of MOEA

Given the fact that different MOEA components determine the search characteristics that affect performance, as long as the general optimisation process is consistent with that outlined in Figure 1-1, different MOEAs can be viewed simply as the set of components used in each process. For example, for NSGA-II, during the selection process, parent selection and survivor selection uses the crowding distance selection strategy; in the reproduction process, NSGA-II uses SBX and PM as the two operators but no hyperheuristic is used. Thus, identifying or fine-tuning the components of these processes would affect the search behaviour and algorithm performance. However, systematic investigations into this topic are not currently sufficient. Given the fact that it is time-consuming to evaluate all of the component combinations, many existing MOEAs' process components have traditionally been determined without understanding their influence on performance. In this work, we considered the best component alternatives found in Chapter 2 (number of operators) and Chapter 3 (selection strategy) and the popular component alternatives in existing MOEAs to identify the best component combination, in terms of the proposed multi-objective evolutionary algorithm [GMOEA(CHC_{Gen},12,T,A)]. The detailed review of the existing methodology for existing MOEA and its limitations is outlined and addressed in Chapter 4.

1.2 Research aims

The main objective of the thesis is to understand the impact of each component of an MOEA on algorithm performance, and to use this knowledge to systematically develop a new MOEA for WDS problems. In order to achieve this, a general multi-objective evolutionary algorithm framework that reflects the general multi-objective evolution algorithm's process, as outlined in Figure 1-1, is proposed. This framework is the algorithmic test bed used to create a fair comparison of the components that enable researchers to isolate the influence of individual components from any other components within the framework. With this framework, in this thesis, three objectives have been proposed, with the specific sub-objectives as shown below.

Objective 1. To study the impact of the size of an operator set on MOEA performance.

Objective 1.1. To assess whether the inclusion of a larger number of operators improves algorithm performance.

Objective 1.2. To assess whether the relative influence of the number of operators (i.e., the size of the operator set) is: greater than that of the composition of this operator set (i.e., which operators constitute this set) (sub-objective 1.2a); and greater than that of the combined effect of other types of strategies affecting the algorithm search (such as parent and survivor selection or the degree to which various operators contribute at different stages of the search) (sub-objective 1.2b).

Objective 1.3. To assess the potential for improving the performance of existing EAs by increasing the size of the operator set within these algorithms.

Objective 2. To study the impact of the selection strategy on MOEA performance.

Objective 2.1. To propose a new convex hull contribution selection strategy for population-based MOEAs.

Objective 2.2. To assess the performance of MOEAs with different selection strategies.

Objective 2.3. To assess the potential for improving the performance of existing MOEAs by using the new convex hull contribution selection strategy.

Objective 3. To propose a new MOEA for WDS problems.

Objective 3.1. To determine the optimal MOEA component configuration (i.e., operator, selection strategy and hyperheuristic) by conducting comprehensive numerical experiments.

Objective 3.2. To investigate the relative influence of each component and pairwise combination of components on algorithm performance.

Objective 3.3. To evaluate the new proposed MOEA's performance by comparing it with seven state-of-the-art MOEAs.

1.3 Organisation of the Thesis

The main body of this thesis (Chapters 2 to 4) comprises the collection of three journal articles produced within this research[2]. A summary of the thesis chapters is given below.

Chapter 2 (Journal paper 1) focuses on investigating the impact of the operator set on MOEA performance. Specifically, a comprehensive numerical comparison has been conducted to study the relative influence of the number

[2] The journal paper manuscripts have been reformatted in accordance with University of Adelaide guidelines, and sections have been renumbered for inclusion within this thesis.

of operators, operator combination, and other combined components on algorithm performance.

Chapter 3 (Journal paper 2) proposes a new convex hull contribution selection strategy. Moreover, the proposed selection strategy was compared with existing selection strategies to study the influence of the selection strategy on algorithm performance.

Chapter 4 (Journal paper 3) proposes a GMOEA by identifying the best performing component combinations. Also, extensive numerical experimentation has been conducted to understand the relative influence of each component and pairwise combination of components on algorithm performance. The findings of this thesis also reinforce the conclusions in **Chapters 2-3**.

Chapter 5 summarises the contributions of the research. Future work is also discussed.

Chapter 2 Do Existing Multiobjective Evolutionary Algorithms Use a Sufficient Number of Operators? An Empirical Investigation for Water Distribution Design Problems

Peng Wang[1,2], Aaron C. Zecchin[1], Holger R. Maier[2,1], Feifei Zheng[2*], Jeffrey P. Newman[1]

[1]School of Civil, Environmental and Mining Engineering, The University of Adelaide, Adelaide, SA, Australia;

[2]College of Civil Engineering and Architecture, Zhejiang University, China

Citation: Wang, P., Zecchin, A. C., Maier, H. R., Zheng, F., & Newman, J. P. (2020). Do existing multiobjective evolutionary algorithms use a sufficient number of operators? An empirical investigation for water distribution design problems. Water Resources Research, 55, e2019WR026031.

Statement of Authorship

Title of Paper	Do Existing Multiobjective Evolutionary Algorithms Use a Sufficient Number of Operators? An Empirical Investigation for Water Distribution Design Problems
Publication Status	☑ Published ☐ Accepted for Publication ☐ Submitted for Publication ☐ Unpublished and Unsubmitted work written in manuscript style
Publication Details	Wang, P., Zecchin, A. C., Maier, H. R.,Zheng, F., & Newman, J. P. (2020). Doexisting multiobjective evolutionaryalgorithms use a suf cient number ofoperators? An empirical investigationfor water distribution design problems.Water Resources Research, 55, e2019WR026031. https://doi.org/10.1029/2019WR026031

Principal Author

Name of Principal Author (Candidate)	Peng Wang		
Contribution to the Paper	Primary innovator, analyst and author Conception and design of the project Development and execution of numerical experimental program Analysis and interpretation of research data Draft the paper		
Overall percentage (%)	85		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	01/10/2020

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Aaron C. Zecchin		
Contribution to the Paper	Conception and design of the project Analysis and interpretation of research data Critically revising the paper so as to contribute to the interpretation		
Signature		Date	17/07/2020

Name of Co-Author	Holger R. Maier
Contribution to the Paper	Conception and design of the project Analysis and interpretation of research data Critically revising the paper so as to contribute to the interpretation

Signature		Date	09/09/2020

Name of Co-Author	Feifei Zheng		
Contribution to the Paper	Conception and design of the project Analysis and interpretation of research data Critically revising the paper so as to contribute to the interpretation		
Signature		Date	03/11/2020

Name of Co-Author	Jeffrey P. Newman		
Contribution to the Paper	Conception and design of the project Analysis and interpretation of research data Critically revising the paper so as to contribute to the interpretation		
Signature		Date	03/11/2020

Please cut and paste additional co-author panels here as required.

Abstract

Multiobjective evolutionary algorithms (MOEAs) have been used extensively to solve water resources problems. Their success is dependent on how well the operators that control an algorithm's search behavior are able to identify near-optimal solutions. As commonly used MOEAs contain a relatively small number of operators (generally between 2 and 7), this chapter investigates whether the performance of MOEAs could potentially be improved by increasing their operator set size. This is done via a series of controlled computational experiments isolating the influence of the size of the operator set (i.e., how many operators are used, ranging from 2 to 12), the composition of the operator set (i.e., which operators are used, given a set number of operators), the search strategy used (e.g., parent selection and survivor selection), and increasing the operator set size of an existing MOEA. These experiments are performed on six benchmark water distribution optimization problems. Results of the 3,150 optimization runs indicate that operator set size is the dominant factor affecting algorithm performance, having a significantly greater influence than operator set composition and other factors affecting algorithm search behavior. In addition, increasing the operator set size of the state-of-the-art MOEA GALAXY, which has been designed specifically for solving water distribution optimization problems, from its currently used value of 6 to 12 increased its performance significantly. These results suggest there is value in investigating the potential of increasing operator set size for a range of algorithms and problem types.

2.1 Introduction

Optimization has been used extensively to solve a wide range of water resources problems for a number of decades (Maier et al., 2014; Mala-Jetmarova et al., 2017, 2018; Nicklow et al., 2010). A persistent thread in this literature is the quest to develop algorithms that perform satisfactorily on the widest range of problem types possible (Maier et al., 2014; Mala-Jetmarova et al., 2017). On the surface, this might appear to be a somewhat utopian pursuit, as it contradicts the no-free-lunch theorem (Wolpert & Macready, 1997), which states that if an algorithm performs better than random search on some class of problems, then it must perform worse than random search on other types of problems. However, over the last two decades, the use of evolutionary algorithms (EAs) has enabled significant progress to be made toward achieving this goal. This is because these algorithms have the ability to adjust the way they search through the solution spaces by either intensifying the search in promising regions (i.e., exploiting good solutions) or diversifying the search in less promising regions (e.g., exploring the solutions space more widely), enabling them to perform well on problems with different characteristics (Maier et al., 2019).

EAs are able to achieve the diverse searching behavior outlined above via the use of a range of operators (Maier et al., 2014; Zecchin et al., 2012; Zheng, 2015; Zheng, Qi, et al., 2017), which provide different strategies for searching through the solution space. For example, crossover operators, such as parent-centric crossover (PCX) (Deb & Agrawal., 1994), simplex crossover (SPX) (Tsutsui et al., 1999), simulated binary crossover (SBX) (Deb & Agrawal., 1994), unimodal distribution crossover (UNDX) (Kita et al., 1999), differential evolution (DE) (Storn & Price, 1997), the adaptive metropolis strategy (AMS) (Haario et al., 2001), and particle swarm optimization (PSO) (Kennedy & Eberhart, 1995) all provide different strategies for intensifying the search in the proximity of high-performing parent solutions. In contrast, operators such as uniform mutation (UM) (Michalewicz, 1992), polynomial mutation (PM) (Deb & Agrawal., 1994), Gaussian mutation (Rechenberg, 1965), the turbulence factor (TF) (Pulido & Coello Coello, 2004), and dither creeping (DC) (Wang et al., 2017) offer different strategies for exploring various regions of the solution space more widely.

The different search strategies provided by various operators can generally be fine-tuned with the aid of one or more parameters. For instance, in PCX and UNDX, the degree of proximity of the search to the parent solutions is controlled by two variance parameters, whereas in SBX this is achieved with the aid of a distribution index, in SPX by a spreading factor, in DE by using a mutation weighting factor and a crossover rate, in AMS by using a jump factor, and in PSO with the aid of a velocity factor. Similarly, in UM, PM, and GM, the degree of exploration is controlled by a probability of mutation, in addition to a distribution index in PM and a scaling factor in GM, whereas three probabilities of mutation parameters are used in DC. In addition to fine-tuning algorithm search behavior by changing the values of these parameters, search behavior can also be changed by dynamically adjusting the degree to which different operators contribute to the identification of better solutions throughout the search based on algorithm performance (Burke et al., 2013).

How many and which operators are used in a particular algorithm is generally fixed, based on the choices made by the developers of these algorithms. For example, NSGA-II (Deb et al., 2002b), which is one of the most commonly and widely used EAs, uses two operators (SBX and PM), whereas the more recently developed algorithms AMALGAM (Vrugt & Robinson, 2007), Borg (Hadka & Reed, 2013), and GALAXY (Wang et al., 2017) use six (SBX, PM, PSO, TF, DE, and AMS), seven (SBX, PCX, SPX, DE, UNDX, UM, and PM), and six (SBX, DE, TF, DC, GM, and UM) operators, respectively. As a result, there has been a large number of studies that have focused on (i) the sensitivity of the performance of these algorithms to values of the operators (Vrugt et al., 2009; Wang et al., 2017), (ii) the impact of adapting values of these operators during the optimization process (Karafotias et al., 2015; Zheng, Zecchin, et al., 2017), and (iii) various strategies for determining the relative contribution of operators at different stages of the search (Burke et al., 2013; Drake et al., 2019).

However, as the use of different operators is akin to the adoption of different search strategies, whereas the adjustment of the parameters that control these operators, and the mechanisms that are used to determine the relative contribution of these operators during the search, akin to fine-tuning the selected strategies, the issue of how many and which operators are included in an algorithm is likely to have a greater influence on algorithm performance than the above factors (as long as reasonable parameter values are selected) (Soria-Alcaraz et al., 2017). Consequently, it is somewhat surprising that the influence of the different numbers of operators on the performance of EAs has received very limited attention, with only Vrugt et al., (2009) exploring this issue. However, they only considered five candidate operators, making it possible to investigate the impact of all possible operator combinations. However, this is not the case when consider a larger number of operators, such as those used in current state-of-the-art algorithms.

In order to address this shortcoming, the overall aim of this paper is to systematically explore the influence of the number of operators on the performance of EAs. The specific objectives are as follows:

1. To assess the relative influence of the size of the operator set on algorithm performance.

2. To assess whether the size of the operator set is more important for algorithm performance than the composition of the operator set (i.e., intentionally constructed operator sets from existing algorithms versus

randomly constructed sets).

3. To assess whether the size of the operator set is more important for algorithm performance than the combined effect of the composition of the operator set and the search strategies used (i.e., intentionally constructed

operator sets and search strategies, such as parent and survivor selection and the use of approaches that govern the usage of operators, from existing algorithms vs. randomly constructed sets).

4. To assess the potential for improving the performance of existing MOEAs by increasing the size of the operator set.

The above objectives are achieved via a large number of computational experiments applied to a range of water distribution system design problems. These problems have been selected as they exhibit a diverse range of problem characteristics (Wang et al., 2015) and have been tested in a number of studies assessing the performance of different optimisation algorithms (Jahanpour et al., 2018; Wang et al., 2015, 2017; Zheng et al., 2016; Zheng, Qi, et al., 2017). In addition, the design of water distribution systems is an important test problem that has been studied extensively in the area of water resources (Mala-Jetmarova et al., 2018). The remainder of this paper is organized as follows. An outline of the methodology is given in section 2.2. The results are presented and discussed in section 2.3, followed by a summary and conclusions in section 2.4.

2.2 Methodology

2.2.1 Background

The general steps in the iterative process by which multiobjective evolutionary algorithms (MOEAs) identify better solutions as the search progresses are shown in Figure 2-1. As can be seen, at the commencement of the optimization process, an initial set of solutions is generated, which is subjected to a selection process to identify better-performing solutions, which have the opportunity to reproduce. The reproduction process results in a set of offspring solutions, which are subject to a survivor selection process to identify the solutions that form part of the next generation. This process of parent selection, reproduction, and survivor selection is repeated until certain termination criteria are met, such as the execution of a fixed number of iterations (generations) or until no better solutions can be identified.

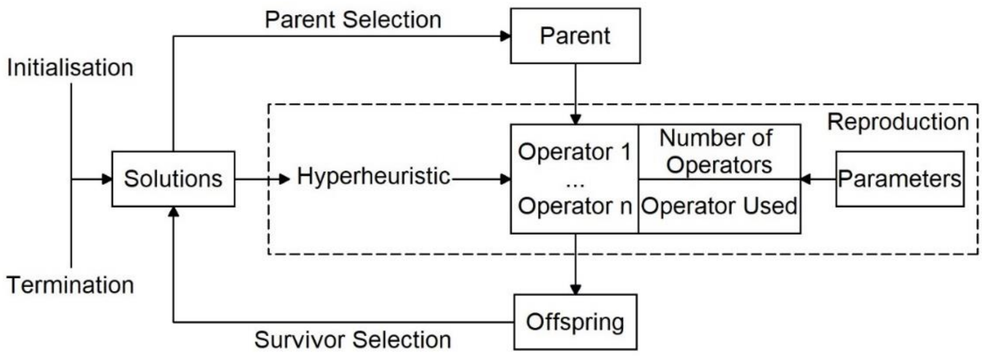


Figure 2-1. Steps in generic multi-objective evolutionary optimisation process

The reproduction process is facilitated by one or more operators or heuristics (Figure 2-1). Consequently, reproduction is affected by the number of operators used (i.e., the size of the operator set) and which operators are used in this set (i.e., the composition of the operator set). As mentioned in section 2.1, the performance of the operators can be fine-tuned with the aid of one or more parameters. In addition, the degree to which each of the operators contributes to the search at each iteration can be controlled with the aid of hyperheuristics, which are high-level automated search methodologies for selecting the most appropriate lower-level operators (or heuristics) (Burke et al., 2013; Drake et al., 2019).

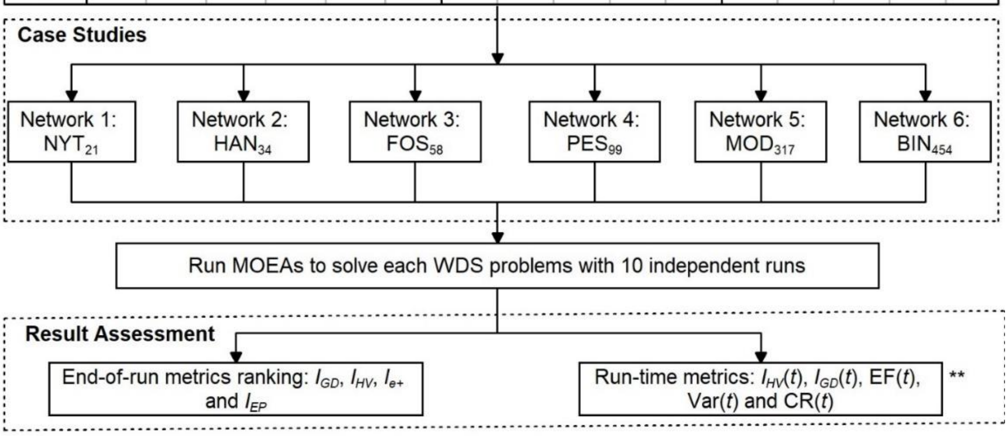
Different MOEAs, such as NSGA-II, GALAXY, or Borg, use different operator set sizes, different operator set compositions, and different search strategies (including different hyperheuristics and parent and survivor selection strategies), making it difficult to isolate the impact of the size of the operator set on algorithm performance, which is the focus of this chapter.

Consequently, a methodology has been developed for achieving this, an overview of which is given in the next subsection.

2.2.2 Overview

In order to enable the relative impact of the inclusion of a larger number of operators on optimization algorithm performance to be assessed, a general MOEA framework was developed (see section 2.3.1 for details). As part of the framework, parent and survivor selection strategies remain fixed, the operator set is varied (according to the type of experiment), and the relative contribution of each operator is held constant throughout the search (i.e., NAÏVE hyperheuristic is used). This enables the influence of the size and composition of the operator set to be assessed in an objective fashion as part of a series of controlled experiments. In order to be able to assess the relative influence of the size of the operator set on algorithm performance (Objective 1), the composition of operator sets of different sizes was randomly generated from a pool of 12 operators via uniform sampling: 10 different combinations of two randomly selected operators; 5 different combinations of 4, 6, and 10 randomly selected operators; and a single set of all 12 operators (Experiments 1a to 1e, Figure 2-2). This minimizes the influence of the composition of the operator set (or any bias induced through intentional construction), thereby isolating the impact of the size of the operator set.

Exp.	Operator Set Size						Operator Set Composition					Search Strategy				
	2	4	6	7	10	12	Random	Constructed 1	Constructed 2	Constructed 3	Constructed 4	Framework	Constructed 1	Constructed 2	Constructed 3	Constructed 4
1a	X*						X[1]					X				
1b		X*					X[2]					X				
1c			X*				X[2]					X				
1d					X*		X[2]					X				
1e						X*	X[3]					X				
2a	X							X				X				
2b	X								X			X				
2c			X							X		X				
2d				X							X	X				
3a	X							X					X			
3b	X								X					X		
3c			X							X					X	
3d				X							X					X
4						X									X	



* Parameter sensitivity analysis for Network 6

** Only for Objective 1

[1] Ten different random combinations of operators of selected size from the available pool of twelve
[2] Five different random combinations of operators of selected size from the available pool of twelve
[3] One random combinations of operators of selected size from the available pool of twelve

Constructed 1: Operator sets (and search strategies) used by NSGA-II
Constructed 2: Operator sets (and search strategies) used by SAMODE
Constructed 3: Operator sets (and search strategies) used by GALAXY
Constructed 4: Operator sets (and search strategies) used by Borg

Figure 2-2. Overview of the methodology adopted to achieve the stated objectives, with the experiment number corresponding to the objective number being addressed

It should be noted that sets that only contained operators with the same search emphasis (i.e., only exploitation or only exploration) were excluded. This resulted in the consideration of 26 unique operator sets. For the sake of simplicity, each group consisting of the same operator set size is referred to as an algorithm group (e.g., “Algorithm Group 2” refers to all algorithms with an operator set size of 2).

Information on the 12 operators used is given in Table 2-1-interested readers should refer to the relevant references within the table for additional information on each of these operators. These operators were chosen as (i) their use has resulted in successful performance on a wide range of test functions (please refer to the references listed in Table 2-1); (ii) many of these operators have been built into state-of-the-art MOEAs, such as Borg and GALAXY; (iii) they exhibit an array of different search behaviors (Hadka & Reed, 2013; Wang et al., 2017) given their different emphasis on exploitation and exploration (Maier et al., 2014); and (iv) algorithms that use these operators have been found to provide at least a satisfactory, if not highly competitive, outcome on a number of WDS problems (Wang et al., 2015, 2017; Zheng et al., 2016). While the parameters affecting the behavior of the operators were set to values suggested in the literature for the majority of the experiments-for example, the parameter values of the constructed operator sets for Borg and NSGA-II were consistent with the recommended settings in the literature (Deb et al., 2002b; Hadka & Reed, 2013), which had also been used in Wang et al., (2015)-the influence of varying the parameters was tested as part of a targeted sensitivity analysis (Figure 2-2) (see section 2.2.5 for details).

Table 2-1. Operators Applied in the Computational Experiments

Operator candidate	Behavioural emphasis
Simulated binary crossover[a] (SBX)	Exploitative
Differential evolution[b] (DE)	Exploitative & Explorative
Parent-centric crossover[c] (PCX)	Exploitative
Unimodal normal distribution crossover[d] (UNDX)	Exploitative
Simplex crossover[e] (SPX)	Exploitative
Polynomial mutation[a] (PM)	Explorative
Uniform mutation for integer[f] (UMI)	Explorative
Gaussian mutation for integers[g] (GMI)	Explorative
Dither creeping for integers[h] (DCI)	Explorative
Differential evolution for integers[b] (DEI)	Explorative & Explorative
Simulated binary crossover for integers[g] (SBXI)	Exploitative
Turbulence factor for integers[i] (TFI)	Explorative

Note: [a]Deb & Agrawal., (1994). [b]Storn & Price, (1997). [c]Deb et al., (2002a). [d]Kita et al., (1999). [e]Tsutsui et al., (1999). [f]Michalewicz, (1992). [g]Wang et al., (2017). [h]Zheng et al., (2013). [i]Pulido et al., (2004).

In order to assess whether the size of the operator set is more important for algorithm performance than the composition of the operator set (Objective 2), the results for algorithms using constructed operator sets that are used in four existing MOEAs (Experiments 2a to 2d, Figure 2-2) were compared with those obtained using randomly generated operator sets (Experiments 1a to 1e). The existing MOEAs from which the constructed operator sets were obtained include NSGA-II, SAMODE, GALAXY, and Borg, which have two, two, six, and seven operators, respectively (Figure 2-2). These MOEAs have been selected as they use different numbers of operators and have been applied successfully to the case studies considered in this paper in previous studies (see section 2.2.3.2 for details).

In order to assess whether the size of the operator set is more important for algorithm performance than the combined effect of the composition of the operator set and the search strategies used (Objective 3), the results for algorithms using constructed operator sets and search strategies that are used in the four existing MOEAs considered, including different parent and survivor selection strategies and the use of different hyperheuristics that govern the usage of operators (Experiments 3a to 3d, Figure 2-2), were compared with those obtained using algorithms that use constructed operator sets but the same search strategy (Experiments 2a to 2d) and randomly generated operator sets (Experiments 1a to 1e).

Finally, the potential for improving the performance of existing algorithms by increasing their operator set size (Objective 4) was assessed by comparing the performance of the best-performing algorithm from Experiments 3a to 3d with that obtained by increasing the number of operators in this algorithm to the largest number of operators considered in this chapter (i.e., a total of 12 operators) (Experiment 4, Figure 2-2).

As shown in Figure 2-2, the above experiments were conducted on six water distribution system (WDS) design case studies, including the New York Tunnel network (NYT), the Hanoi network (HAN), the Fossolo network (FOS), the Pescara network (PES), the Modena network (MOD), and the Balerma irrigation network (BIN), minimizing network cost and maximizing network resilience for each (see section 2.2.4 for details). These case studies were selected, as they have different levels of complexity and were included in a benchmarking study of different MOEAs for this problem type by Wang et al., (2015). All optimization runs were repeated from 10 different starting positions in decision variable space (random seeds), as shown in the central block of Figure 2-2, resulting in a total of 3,150 optimization runs.

To enable the results from the different computational experiments to be compared in an objective fashion and to understand the reasons for the relative performance of different algorithms, a range of metrics were used, as shown in the "Result Assessment" block in Figure 2-2. The performance rank of the different algorithms used in each experiment was determined by applying the one-way Kruskal-Wallis test (Kruskal & Wallis, 1952) with Dunn's D posttest (Dunn, 1964) to a number of end-of-run performance metrics. In addition, a number of run-time metrics were used to explore and better understand the impact of operator set size and composition on algorithm searching behavior and performance (see section 2.2.6 for details).

2.2.3 Optimization Algorithms

2.2.3.1 General MOEA Framework

As mentioned previously, the general MOEA framework was developed to enable the influence of different operator set sizes and compositions to be tested in an unbiased fashion. The general MOEA framework follows the general structure of NSGA-II, with the addition that it allows for an arbitrary operator set, where the use of the operators is governed by the NAÏVE hyperheuristic (i.e., equal computational resources are allocated to each operator). The reason for basing the structure of the general MOEA framework on that of NSGA-II is that its simplicity allows for the resulting search performance to largely be attributed to the choice of operator set, which is the purpose of this chapter, and that it forms the basis of the structure of a number of other popular MOEAs, such as AMLAGAM and GALAXY.

The pseudo code of the framework is shown in Figure 2-3. As can be seen, the inputs to the framework are population size (N), number of function evaluations (N_{NFE}), and the operator set $\boldsymbol{\Theta} = \{\Theta_1, \ldots, \Theta_k\}$, consisting of k operators, where Θ_j is the j-th operator (line 1). Starting from the initialization (line 2), a population $\boldsymbol{x}$ of N solutions is uniformly sampled from the sample space and ranked by the fast non-dominated sorting approach (Deb et al., 2002b). In the main loop (lines 3 – 10), the parent solutions are selected from the population set by implementing the constraint tournament selection strategy (Deb et al., 2002b). As part of this strategy, the infeasible solutions were recognized as dominated by all feasible solutions and ranked in ascending order based on degree of constraint violation. Solutions with the highest degree of constraint violation had the smallest probability of being selected.

As part of the reproduction process (lines 5 - 7), the operator set $\boldsymbol{\Theta}$ is used to produce offspring solutions. A key feature of the framework is that the size, k, and composition of the operator set $\boldsymbol{\Theta}$, are variable, so that the influence of the size and composition of the operator set on algorithm performance can be assessed in an objective fashion. It should be noted that the quota q_j for each operator in $\boldsymbol{\Theta}$ is equal, indicating that no biasing hyperheuristics are applied. For each operator, after q_j offspring $\mathbf{y}_j$ are produced, they are added to the offspring set $\boldsymbol{y}$ (line 6). After evaluation of the solutions in the offspring set, the $\boldsymbol{y}$ are combined with $\boldsymbol{x}$ (line 9). In the survivor selection step, the crowding distance replacement strategy (Deb et al., 2002b) is used to select the population for the next generation (line 10) from $\boldsymbol{y} \cup \boldsymbol{x}$. In addition, for the infeasible solutions, the solutions with a higher violation of the constraints would be less likely to be included into the population set. The main loop is terminated if the current i_{NFE} is greater than the total N_{NFE} (line 3).

Inputs: population size (N), number of function evaluations (N_{NFE}), and operator set $\boldsymbol{\Theta} = \{\Theta_1, \ldots, \Theta_k\}$
Initialize N individuals as the population set $\boldsymbol{x}$; the quotas of each search operator $q_j = \left\lfloor \frac{N}{k} \right\rfloor$; $i_{\text{NFE}} = 0$
while $i_{\text{NFE}} <= N_{\text{NFE}}$
    Select parent solutions, set $\boldsymbol{y} = \varnothing$
    for $j = 1$ to k
        Produce q_j offspring solutions $\boldsymbol{y}_j$, and add to $\boldsymbol{y}$
    end
    Evaluate $\boldsymbol{y}$
    Combine the $\boldsymbol{x}$ and $\boldsymbol{y}$
    Implement replacement strategy to select the survival individual to form the population set
    $i_{NFE} = i_{NFE} + \sum_{j=1}^{k} q_j$
Loop
end
Outputs: Pareto approximation set ($\boldsymbol{z}$), Pareto approximation front ($\boldsymbol{f}(\boldsymbol{z})$)
end

Figure 2-3. Pseudocode for the general MOEA framework used to test the relative influence of operator set size and composition

2.2.3.2 Existing MOEAs

As mentioned in section 2.2.2, for the purposes of benchmarking, the constructed operator sets used in Experiments 2a-2d and the constructed operator sets and search strategies used in Experiments 3a to 3d are those used in four existing MOEAs: NSGA-II, SAMODE, GALAXY, and Borg. These constructed operator sets and search strategies have been selected as their source MOEAs (i) are constructed from different operators set sizes (two for NSGA-II, two for SAMODE, six for GALAXY, and seven for Borg), compositions and search strategies (e.g., hyperheuristics and parent and survivor selection strategies), and (ii) have been applied successfully to the case studies considered in this research previously (e.g., Wang et al., 2015, 2017; Zheng et al., 2016).

NSGA-II is recognized as an industry standard MOEA for WDS optimization problems and has the simplest structure of the algorithms considered in this chapter. It also has been found to outperform more advanced MOEAs, such as Borg, in terms of the contribution percentages to the best-known Pareto front and end of runs metrics (i.e., hypervolume and generational distance), for the case studies considered in this paper (Wang et al., 2015; Zheng et al., 2016). In these studies, NSGA-II has shown faster convergence to the best-known

Pareto fronts and also maintained better solution diversity. Borg is a robust hybrid MOEA that has been shown to perform well on a wide range of applications and test functions (Hadka & Reed, 2012, 2013, 2015; Zeff et al., 2016), and SAMODE is a parameter adaptive MOEA that has been shown to perform well for a range of WDS problems (Zheng et al., 2014, 2016). GALAXY was designed specifically for solving WDS problems and has been shown to outperform many other MOEAs for the WDS problem case studies considered in this paper. A brief description of each of these MOEAs is given below.

NSGA-II (Deb et al., 2002b) combines fast nondominated sorting, elitist preservation (based on Pareto dominance rank), and filtering of the population's solutions based on crowding distance. The algorithm contains two operators (both used for every offspring construction): the crossover operator SBX and the mutation operator PM. These operators were selected based on their robust combined performance across a wide range of test functions (Deb & Agrawal., 1994).

SAMODE was first proposed by Zheng et al., (2014). It combines fast nondominated sorting and use of a crowding distance to maintain population convergence and diversity, as is the case with NSGA-II. The parameters used to fine-tune operator search behavior, including the mutation weighting factor, F, and the crossover rate, CR, are embedded into each solution string. The values of the parameters are adaptive: If a given set of parameter values results in offspring solutions that dominate their corresponding parent solutions, these parameter values are retained in the next generation; otherwise, they are randomly generated within pre-specified ranges.

Borg (Hadka & Reed, 2013) is a hybrid MOEA that utilizes multiple operators and combines a number of advanced search strategies, such as ε-progress, ε-dominance archive, randomized restart, and adaptive tournament selection size. Moreover, Borg uses a hyperheuristic to auto-adapt its operators, where a feedback loop is established in which operators that produce greater successful offspring are rewarded by increasing their selection probabilities for producing offspring solutions in subsequent generations.

GALAXY (Wang et al., 2017) is a hybrid MOEA that is tailored to discrete combinational problems, such as the WDS design problem. It uses six operators that have been selected based on empirical and preliminary numerical investigations, and for the purpose of facilitating a diverse array of search behaviors. In addition, several advanced strategies are built into GALAXY to improve algorithm performance, such as hybrid replacement, the global sharing strategy, the duplicates handling strategy, and a hyperheuristic called genetically adaptive strategy.

2.2.3.3 Implementation

The code for the general MOEA framework was developed using MATLAB's M-script, and the operator codes were developed to be consistent with the code of NSGA-II (Deb et al., 2002b), Borg (Hadka & Reed, 2013), and GALAXY (Wang et al., 2017). The code for implementing NSGA-II was obtained from Deb et al., (2002b); the code for implementing SAMODE was obtained from Zheng et al., (2016); the code for implementing GALAXY was obtained from Wang et al., (2017); and the code for implementing Borg was obtained from Hadka and Reed (2013). For all experiments, EPANET 2.0 (Rossman, 2000) was used to perform the hydraulic simulations needed to evaluate the pressures at each node of the WDSs considered. All optimization runs were conducted on the Phoenix High Performance Computer (HPC) at the University of Adelaide. Phoenix HPC is a heterogeneous hardware system that includes a mix of CPU-only and CPU/GPU-accelerated nodes. It has 260 nodes in total, which equipped with 2X Intel Gold 6148, 40 cores @ 2.4 GHz, 384 GB memory for CPU nodes. In addition, the max RAM per node is 125 GB.

2.2.4 Case Studies

2.2.4.1 Formulation of Optimization Problem

WDS optimization is an NP-hard combinatorial problem that is nonconvex, high dimensional, multimodal, and nonlinearly constrained (Zecchin et al., 2012). The problem can be defined as the selection of the lowest-cost combination of appropriate component sizes and component settings, such that demands and other design constraints are satisfied. A common bi-objective formulation of the problem, adopted by many authors (Bi et al., 2016; Jahanpour et al., 2018; Wang et al., 2015, 2017; Zheng et al., 2016), is the consideration of the cost and resilience of a network as the two objective functions. Therefore, the maximization of network resilience (Prasad & Park, 2004) and minimization of network cost are considered as the two objectives in this chapter. The cost objective is given by

$$F_c = a\sum_{i=1}^{n} D_i^b L_i \tag{2-1}$$

where F_c = total network cost, which is determined by pipe diameter D_i and pipe length L_i; a and b = specified cost function coefficients; n = total number of pipes in the network. The network resilience objective, which measures of combined effect of surplus power and nodal uniformity, is given by:

$$I_n = \frac{\sum_{j=1}^{m} U_j Q_j (H_j - H_j^*)}{\sum_{j=1}^{N_R} Q_{r,j} H_{r j} - \sum_{j=1}^{m} Q_j (H_j^* + z_j)} \tag{2-2}$$

where I_n = the network resilience, for which the numerator represents the surplus power combined with the nodal uniformity for all of the nodes and for which the denominator indicates the maximum surplus power; m = total number of demand nodes; Q_j, H_j and H_j^* are, demand, actual head, and minimum head required at each node j, respectively; $N_R =$ total number of reservoirs; $Q_{r,j}, H_{r,j}$ are the actual discharge and actual head at reservoir j; and U_j is an indicator of diameter uniformity for pipes that are connected to node j and is defined as:

$$U_j = \frac{\sum_{i=1}^{N_{p,j}} D_{ij}}{N_{p,j} \max\{D_{ij} : i = 1, \ldots, N_{p,j}\}} \tag{2-3}$$

where D_{ij} = the diameter of the i-th pipe connected to node j; $N_{p,j} =$ total number of pipes that are connected to node j. Note that a larger value of U_j represents a higher reliability of the network node since the diameter variations between these pipes are lower overall (U_j =1 when all pipe diameters are identical) (Prasad & Park, 2004).

In this chapter, the pipe size decision variables were encoded as consecutive integer values, ranging from one to the number of commercially available sizes. The constraints of the WDS optimization problem were flow velocity in each pipe and pressure head at each node. The satisfaction of the constraints, or otherwise, was computed by the hydraulic simulation software EPANET 2.0 (Rossman, 2000).

2.2.4.2 Case Studies

The six case studies considered have been used in a number of previous studies assessing the relative performance of different MOEAs (Wang et al., 2015, 2017; Zheng et al., 2016) and represent problems with different characteristics (Table 2-2). The size of the case study networks varies considerably, with the number of pipes ranging from 21 to 454 (Table 2-2), corresponding to problems with a range of characteristics (Wang et al., 2015). The population size N and computational budgets (i.e., NNFE) used are consistent with those used in Wang et al., (2017), who selected these values based on the results of a number of computational experiments considering a range of population sizes and MOEAs.

As can be seen from Table 2-2, both population size and computational budget are related to network size (i.e., the number of pipes). In terms of population size, values of 100 were used for the small- and medium-sized NYT, HAN, FOS, and PES problems, whereas values of 200 were used for the larger MOD and BIN problems. In terms of the number of function evaluations, values of NNFE = 50,000, 100,000, and 400,000 were used for the medium, intermediate, and large-scale WDS case studies, respectively.

Table 2-2. WDS Case Studies and Population Size of the MOEAs

Scale	Case study (problem)	Number of options for each pipe	N_{NFE}	N
Small	New York tunnel (NYP_{21})	16	$5x10^4$	100
	Hanoi (HAN_{34})	6	$5 x10^4$	100
Intermediate	Fossolo (FOS_{58})	22	$1 x10^5$	100
	Pescara (PES_{99})	13	$1 x10^5$	100
Large	Modena (MOD_{317})	13	$4 x10^5$	200
	Balerma (BIN_{454})	10	$4 x10^5$	200

2.2.5 Parameter Sensitivity Analysis

The values of the parameters for the 12 operators used (Table 2-1) were those recommended by Wang et al., (2015, 2017). However, in order to ensure the results obtained are robust, the sensitivity of the relative performance of the algorithms to the choice of parameter values was tested on one instance of each algorithm group for the Balerma network (BIN), which is the largest of the case study networks considered (Figure 2-2). The search behavior of the 12 operators considered is affected by 11 parameters (Table 2-3). For each of these, three choices were considered, including the recommended value and a ±10% deviation from this. This is considered appropriate, as guidelines for reasonable values of these parameters are well established.

Table 2-3. Sampled parameter ranges for the uncertainty analysis

Parameter	Parameter values		
	Min	Recommended	Max
SBX rate	0.81	0.9	0.99
SBX Distribution index	13.5	15	16.5
DE Crossover rate	0.09	0.1	0.11
DE Differential weight	0.45	0.5	0.55
UNDX Zeta	0.45	0.5	0.55
UNDX Eta	0.315	0.35	0.385
PCX Eta	0.09	0.1	0.11
PCX Zeta	0.09	0.1	0.11
SPX Expansion rate	2.7	3	3.3
PM Rate	0.9*1/n	1/n	1.1*1/n
PM Distribution index	0.63	7	7.7

Details of the number of parameters that are relevant for each of the algorithms with different operator numbers included in the sensitivity analysis are given in Table 2-4. In order to ensure that representative combinations of all possible parameter combinations were included in the sensitivity analysis for each algorithm, the space of possible parameter combinations was sampled using a Latin hypercube approach. The number of samples used for each algorithm was calculated as "the number of options for each parameter" (i.e., 3) times "the number of parameters included in the algorithm," based on the suggestion by Munoz and Smith-Miles (2017). This resulted in a total of 105 different parameter combinations (Table 2-4). As each optimization run was repeated 10 times from different random starting positions in the decision variable space, the parameter sensitivity analysis consisted of an additional 1,050 optimization runs.

Table 2-4. Number of Latin hypercube samples used for algorithms with different numbers of operators as part of the parameter sensitivity analysis

Algorithm Instance	Number of Parameters	Number of Parameters Options	Sample Size
Algorithm 2	2	3	6
Algorithm 4	7	3	21
Algorithm 6	4	3	12
Algorithm 10	11	3	33
Algorithm 12	11	3	33
Total			105

2.2.6 Result Assessment

2.2.6.1 End-of-Run Performance Metrics

Four end-of-run metrics were used to assess the relative performance of the different algorithms investigated, as they are able to capture both the best-known solutions' convergence and diversity. The four metrics are hypervolume (I_{HV}) (Zitzler & Thiele, 1999), generational distance (I_{GD}) (Veldhuizen, 1999), ε-indicator ($I_{\varepsilon+}$) (Zitzler et al., 2003) and ε-performance (I_{EP}) (Kollat & Reed, 2005). I_{HV} measures both convergence and diversity of solutions by computing the ratio of the volume of a set of solutions to that of the best-known Pareto front, relative to a fixed reference point. I_{GD} measures convergence of solutions by calculating the average distance between a set of solutions and the best-known Pareto front. $I_{\varepsilon+}$ is designed to measure the convergence and consistency of a solution set by computing the minimum distance required to shift this set to dominate the best-known Pareto front. I_{EP} evaluates the proportion of solutions that is within a user-specified ε-value from the best-known Pareto front (the ε-value applied in this chapter is consistent with that used in Wang et al., 2015). Therefore, greater values of I_{HV} and I_{EP} and smaller values of I_{GD} and $I_{\varepsilon+}$ indicate better performance.

Where required, the best-known Pareto fronts for the case studies considered were those obtained by Wang et al., (2015).

Additionally, to enable each existing algorithm/algorithm group to be ranked in a statistically robust fashion, the one-way Kruskal-Wallis test (Kruskal & Wallis, 1952) (with Dunn's D post-test (Dunn, 1964)) was implemented to calculate the statistical significance of the differences in performance between each existing algorithm/algorithm group for each of the four metrics described above. The Kruskal-Wallis test is a nonparametric test that is used for comparing the statistical properties of two or more independent samples, of equal or different sample sizes, to determine if their medians are significantly different. If the nonparametric analysis is significant, Dunn’s D post-test is used to determine if one algorithm performs significantly better than another as part of a pairwise comparison. This analysis enables the rank of each algorithm to be determined by counting the number of times an algorithm performs better, and the number of times it performs worse, than the rest of the algorithms considered. This statistical test provides a robust indication that any observed differences are not as a result of random chance (Ameca-Alducin et al., 2018; Hadka & Reed, 2012).

2.2.6.2 Run-time metrics

In order to better understand the impact of different operators on the ability to find good regions within the search space, and ultimately the best-known Pareto front, the relative operator contribution rate, $CR(t)$ (Eq. 2-4), is introduced. This metric consists of the fraction of successful offspring produced by each operator j in each generation t, which is based on the concept of measuring the contribution of the offspring solutions to the population set (Vrugt & Robinson, 2007). The time-varying nature of this metric enables the relative contribution of different operators to the identification of the set of best-known solutions to be determined at different stages of the optimisation process.

$$CR_j(t) = \frac{k}{N} \cdot P_j^t \tag{2-4}$$

where P_j^t is the number of solutions produced by operator *j* that contribute to the population of best-known solutions at generation *t*.

To better understand the impact of different operators on algorithm performance, the *CR*(*t*) values were not only calculated for individual operators, but also for combinations of operators that contribute to a different behavioural emphasis. Accordingly, referring to the categorisation of each operator in Table 2-2, the total *CR*(*t*) for exploitation / exploration / exploitation and exploration can be estimated by summing up the *CR*(*t*) values of each operator with the same behavioural emphasis. This allows an investigation of the impact of operators that result in exploitative and explorative behaviour on algorithm performance.

To enable *CR*(*t*) values at different stages of the search to be related to algorithm performance, a number of run-time algorithm performance metrics were used, including the Average Euclidean distance (*AED*(*t*)), the hypervolume indicator (*HI*(*t*)), and the extent of the front (*EF*(*t*)). *AED*(*t*) measures the distance of the population of solutions to the reference set, in terms of $I_{GD}(t)$; *HI*(*t*) measures the generation-wise value of $I_{HV}(t)$ for the current population of solutions, and *EF*(*t*) measures the extent of the non-dominated front in the objective domain (Zheng et al., 2016). In addition, in order to understand the impact of different operator contributions on the nature of the solutions, the averaged population variance (*Var*(*t*)) was used, which measures the mean population solution variance in decision variable space (Zheng et al., 2016).

2.3 Results and Discussion

2.3.1 Relative Influence of the Size of the Operator Set

2.3.1.1 Performance Comparison

The rankings of the different algorithm groups considered in Experiments 1a to 1e with respect to the four end-of-run performance metrics, as well as their average rank, are shown in Figure 2-4 as categorical surface plots. In Figure 2-4 (a) – (d), the case studies are given along the horizontal axis (in order of increasing complexity) and algorithm groups (experiments) are given on the vertical axis (ordered by operator set size). The shade in each box indicates the rank of each algorithm group for the given case study (where a darker shade indicates a lower ranking– for example, Algorithm group 6 is ranked as the third best algorithm group for the FOS case study according to the I_{HV} metric). In Figure 2-4 (e), the columns are associated with the different metrics, and each block is the average rank value across the six case studies for each algorithm.

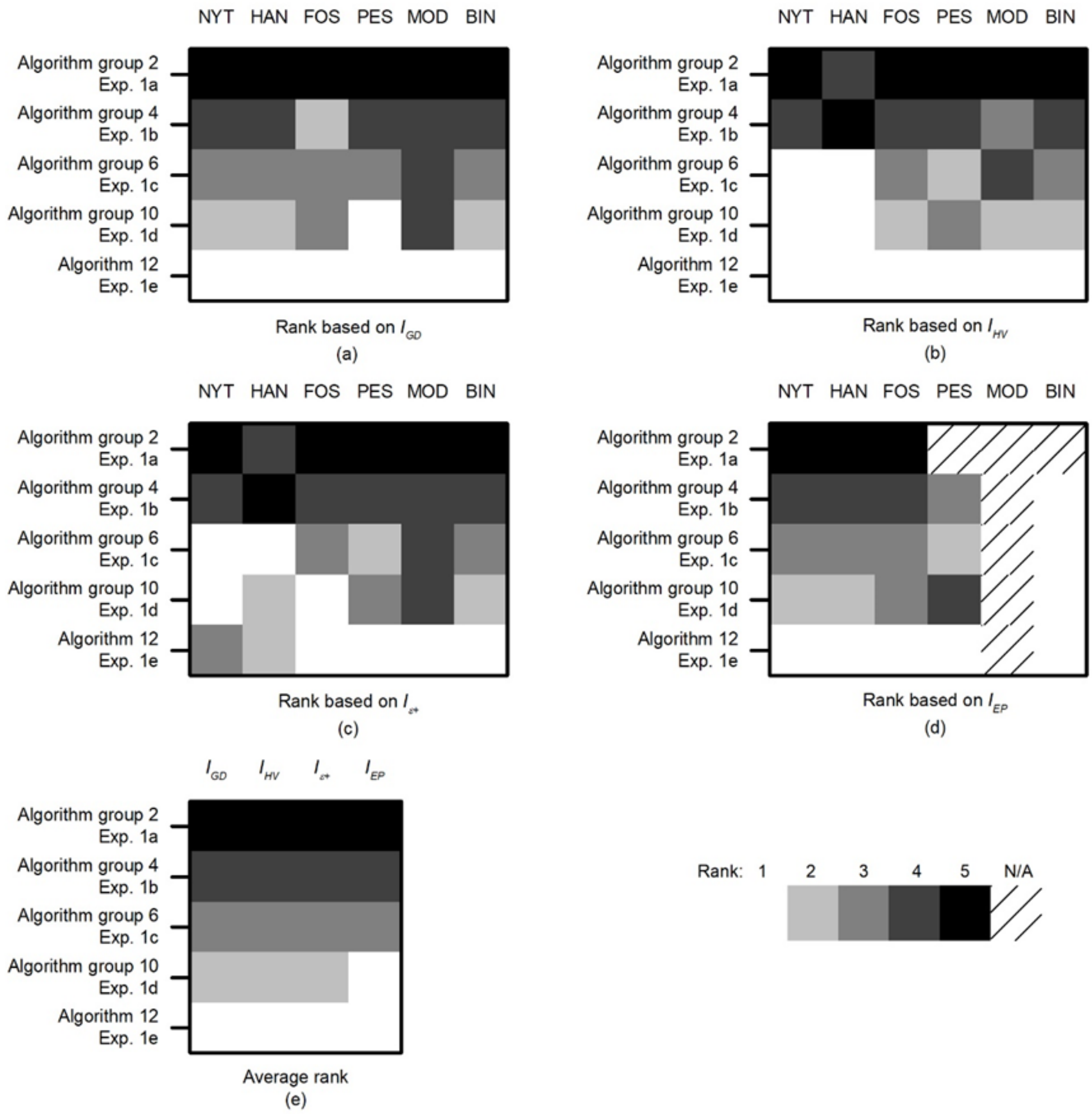


Figure 2-4. Ranks of the four end-of-run performance metrics for each algorithm group in Experiments 1a to 1e across the six WDS case studies. Note: Kruskal-Wallis with Dunn D tests are used to check for statistical differences in the four metrics across 10 duplication runs with different random seeds; subfigures are: (a) rank based on I_{GD}; (b) rank based on I_{HV}; (c) rank based on $I_{\varepsilon+}$; (d) rank based on I_{EP}; and (e) Overall average ranks; the sparse line in (d) means all algorithms failed to capture the solutions near the best-known Pareto fronts.

From Figure 2-4, it can be consistently observed that across all four end-of-run performance metrics, an algorithm groups' rank improves as the size of its operator set increases for this type of problem. For example, in Figure 2-4 (e), the average rank of an algorithm group improves as the size of its operator set increases, with Algorithm group 10 and Algorithm 12 achieving the best average rank (I_{GD} and I_{HV}). In contrast, Algorithm group 2 maintains the worst average rank across all six case studies. Consequently, these results suggest that increasing the operator set size is able to improve an algorithm's performance.

In addition, the improved performance for the larger algorithm groups is emphasised as problem complexity increases. It can be seen from Figure 2-4 (a) - (d) that Algorithm groups 6, 10 and Algorithm 12 typically have similar ranks for the smaller scale problems (e.g. NYT and HAN). However, for the larger scale problems (e.g. MOD and BIN), Algorithm group 10 and Algorithm 12 perform significantly better than all other algorithm groups. It should be noted that in Figure 2-4 (d), the I_{EP} ranks are not available for the MOD problem as none of the algorithms successfully identified solutions close to the best-known Pareto front within epsilon precision, which is identical to the results in Wang et al., (2017). The effectiveness of using Algorithm group 10 and Algorithm 12 is confirmed by their typically higher ranking in terms of I_{HV} for the larger case studies from Figure 2-4 (b). This finding implies that the adoption of larger operator sets yields greater benefits when dealing with problems of higher complexity.

The results of the parameter sensitivity analysis typically show that the variation in parameters considered had little impact on algorithm rankings (Figure 2-5), indicating that the influence of the number of operators outweighs the influence of parameter values. Consequently, the overall conclusion with respect to objective 1 is that when the influence of the number of operators is isolated, there is a statistically significant increase in algorithm performance with an increase in the operator set size.

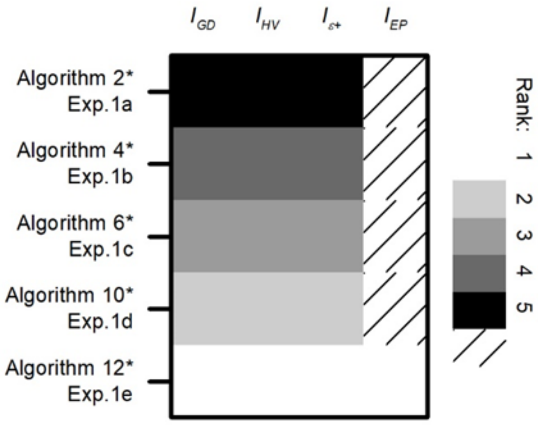


Figure 2-5. Ranks of the selected operator sets with varied/recommended parameter values for BIN problems; Algorithms 2*, 4*, 6*, 10*, 12* use the varied parameter values.

2.3.1.2 Discussion

For the sake of illustration, the results for Algorithm 12 and for three of the randomly selected operator combinations for Algorithm Group 4 (a, b, and c—see the table in Figure 2-7b) are presented and discussed for the BIN problem.

The impact of including operators that focus on either exploration or exploitation is clearly demonstrated by comparing the total $CR(t)$ values for exploitation and exploration (Figure 2-6) and the resulting Pareto fronts (Figure 2-7) for Algorithm group 4a and 4b. These algorithms were selected because they are dominated by exploitative (i.e. three exploitative operators and one operator exhibiting both explorative and exploitative behaviour), and explorative (i.e. two explorative operators, one exploitative operator and one operator exhibiting both explorative and exploitative behaviour) operators, respectively. As can be seen by comparing Figure 2-6 (a) and (b), for Algorithm group 4a, the contributions to the non-dominated solutions are dominated by the exploitative operators, while the reverse is true for Algorithm group 4b.

For Algorithm group 4a, the *CR*(*t*) values in the initial search phase are greater than those for Algorithm group 4b, as the exploitative operators are able to find good regions of the search space relatively quickly (as shown by a comparison of the yellow best-known solutions to the best-known Pareto front, after only 2,000 NFEs – see Figure 2-7 (a)). However, as can be seen in this figure, the extent of the front of the best-known solutions is limited, focusing on the low-cost region of the front. Due to the limited explorative capacity of this algorithm, the *CR*(*t*) values drop off quickly to values around 0.5, indicating that it is more difficult to find improved solutions as the search progresses. This is highlighted by the fact that the extent of the front of best-known solutions for this algorithm is unable to expand by the end of the search (Figure 2-7 (b)) and is only able to move closer to the best-known Pareto front in the low-cost region of the solution space.

For Algorithm group 4b, the *CR*(*t*) values in the initial search phase are smaller than those for Algorithm group 4a, as the exploration dominated search of this algorithm takes longer to find improved solutions. However, as can be seen in Figure 2-7 (a), this emphasis on exploration means that while the resulting front of best-found solutions is not as close to the best-known Pareto front solutions in the early stages of the search, the extent of the front of best-known solutions is much greater. While there is also a reduction in *CR*(*t*) values for this algorithm as the search progresses, this drop is far less pronounced than for Algorithm group 4a, with *CR*(*t*) values stabilising at values of around 1. This continued exploration of the search space results in the front of best-known solutions moving closer to the best-known Pareto front as the search progresses, producing a front that is quite close to the best-known Pareto front at the end of the search (Figure 2-7 (b)).

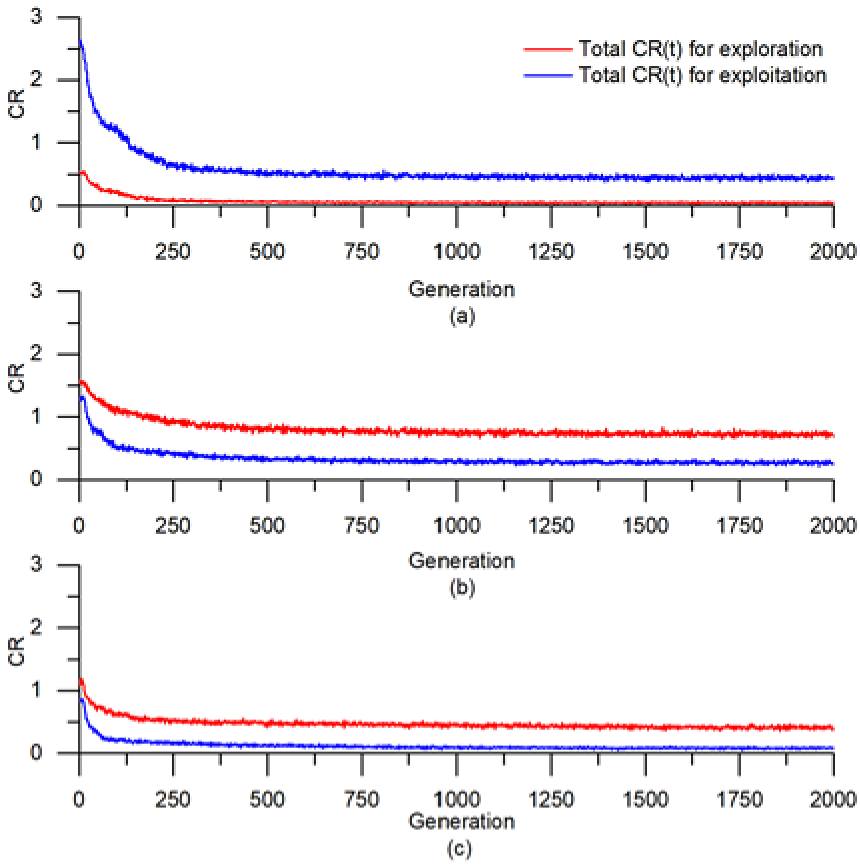


Figure 2-6. Total CR(t) for exploitation and exploration of the selected algorithms for the BIN problem. Note: Each line is the average value over 10 different runs. Note subfigures are: (a) Algorithm group 4a; (b) Algorithm group 4b; and (c) Algorithm group 4c.

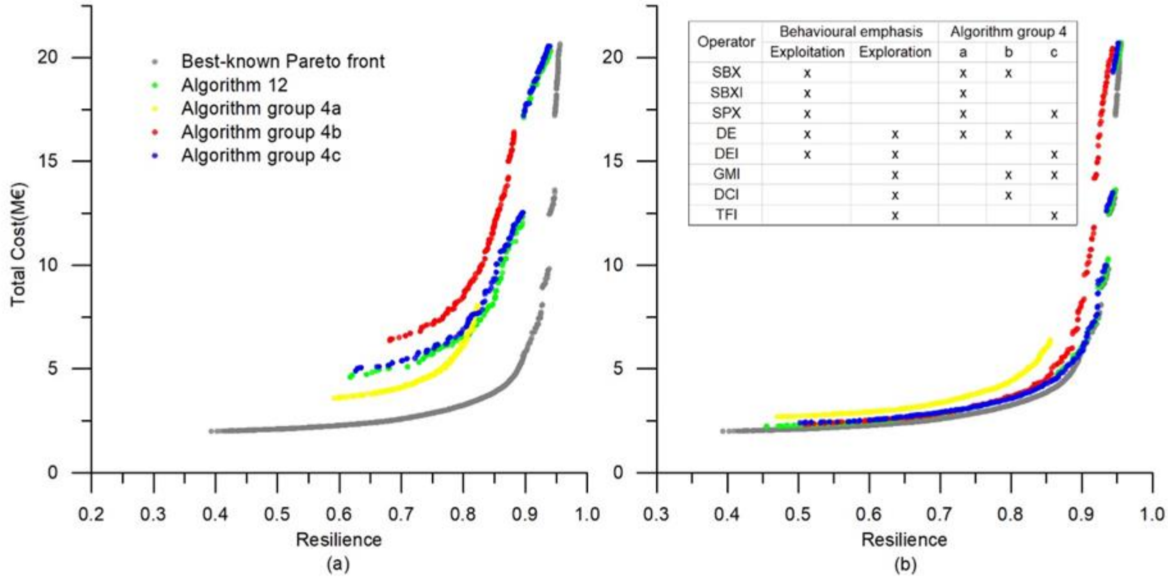


Figure 2-7. Pareto front solutions of Algorithm group 4 and Algorithm 12 at different search stages for the BIN problem. Note subfigures are: (a) is the best-known solutions at iNFE =2,000; and (b) the Pareto front solutions at iNFE =400,000.

The relationship between operators, search behaviour and solution quality described above is consistent with our knowledge of the shape of the fitness landscape for typical WDS design problems. Bi et al., (2016) found that this landscape is shaped like a "big bowl" structure, where:

(i) there are regions in the solution space far from the near-optimal region where rapid improvements in objective function values occur, thereby favouring exploitative operators in the initial stages of the search; and

(ii) there is a large region of the search space (a near-optimal region) where changes in objectives with changes in solutions are relatively small, thereby favouring explorative operators once this region of the fitness landscape has been reached.

In addition to demonstrating the impact explorative and exploitative operators have on algorithm search behaviour, the results also highlight the importance of the *nature* of the explorative or exploitative behaviour induced by different operators. For example, Algorithms 4b and 4c both have two explorative operators, one exploitative operator and one operator exhibiting both explorative and exploitative behaviour, resulting in $CR(t)$ values that are dominated by explorative behaviour for both algorithms (Figure 2-6 (b) and (c)). However, the fronts of the best-known solutions obtained from these two algorithms are very different. Algorithm group 4c is able to not only achieve a greater extent of the front of best-known solutions, but also to move closer to the best-known Pareto front in comparison to Algorithm group 4b, within both the initial and final stages of the search (Figure 2-7 (a) and (b)). This indicates that the search behaviour of the two algorithms is very different, where, collectively, the different sets of operators used in Algorithm group 4c result in greater exploration and exploitation of the search space compared with the set of operators used in Algorithm group 4b.

The differences in the *nature* of the collective explorative and exploitative search behaviour of Algorithms 4b and 4c can be explained by examining the individual search behaviours of the particular operators used in these two algorithms. While both algorithms use the GMI operator as one of their explorative operators, Algorithm group 4b uses DCI as its other explorative operator, while Algorithm group 4c uses TFI. In addition, Algorithm group 4b uses DE, which results in both explorative and exploitative search behaviour, whereas Algorithm group 4c uses DEI, which exhibits similar search behaviours to DE. Finally, Algorithm group 4b uses SBX as an exploitative operator, while Algorithm group 4c uses SPX. Given that both algorithms use GMI and that the behaviour of DE and DEI is similar, the differences in search behaviour of Algorithms 4b and 4c has to be due to the differences in the search behaviour in the other operators (i.e. DCI vs. TFI and SBX vs. SPX).

The differences in the way DCI, TFI, SBX and SPX generate solutions are illustrated in Figure 2-8 as a 2-D plane. As can be seen, DCI tends to explore a small area next to a parent solution (Figure 2-8 (a)), whereas TFI is able to explore areas that are further away from a parent. In other words, TFI has a greater exploration capacity than DCI. This results in Algorithm group 4c being able to cover a greater extent of the front of best-known solutions in comparison to Algorithm group 4b (Figure 2-7). For the exploitative operators, SBX tends to exploit in directions that are orthogonal to the parent solutions (Figure 2-8 (b)), whereas SPX focuses on exploiting a simplex area that is defined by the parent solutions (Figure 2-8 (d)). As seen in Figure 2-8 (b), SPX is able to potentially produce offspring solutions along oblique directions next to each parent solution (i.e. the diagonal directions in Figure 2-8 (b)), whereas SBX only produces solutions that are aligned along the decision variable axes (i.e. orthogonal directions only). Thus, SPX has a greater capacity for exploitation than SBX. This ability to exploit a larger region about the parental solutions results in Algorithm group 4c's best-known solutions moving closer to the best-known Pareto front compared with those of Algorithm group 4b (Figure 2-7). Consequently, even though the search of both Algorithms 4b and 4c is dominated by explorative operators, the greater explorative capacity of Algorithm group 4c results in a higher diversity in the objective space (Figure 2-7), which is due to the difference in the *nature* in which the operators search the solution space.

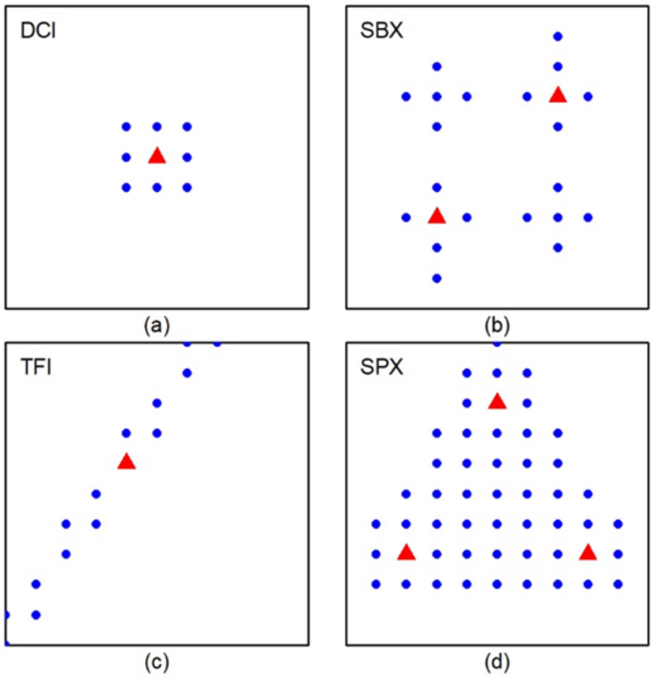


Figure 2-8. Illustration of offspring distributions of different operators in discrete space, where the parents are indicated by the red triangle. Note subfigures are: (a) DCI, (b) SBX, (c) TFI and (d) SPX.

The above results highlight the importance of the *nature*, not just the *behavioural emphasis*, of the way different operators modify solutions from one generation to the next. However, it should be noted that this difference in the nature of the searching behaviour of algorithms with the same behavioural emphasis is not able to be captured by the *CR(t)* values. For example, the *CR(t)* values for Algorithm group 4b (≈1) are greater than those for Algorithm group 4c (≈0.5) throughout the second stage of the search (Figure 2-6 (b) and (c)). This highlights that although Algorithm group 4b is able to identify a larger number of better solutions from one generation to the next, these improvements are not due to the extension of the front. This demonstrates that although *CR(t)* values provide an indication of the number of improved solutions that are identified at each generation, they do not provide an indication of the type and extent of the improvement achieved.

The value of the inclusion of operators with different types of exploitative and explorative search behaviours is demonstrated clearly by the *CR*(*t*) values for Algorithm 12, which consists of five exploitative operators, five explorative operators and two operators that exhibit both explorative and exploitative behaviour (Figure 2-9 (a)-(c)). As can be seen, even though certain operators have a larger contribution towards the identification of better solutions throughout the search, all 12 operators provide at least *some* contribution towards the identification of better solutions at *all* stages of the search. This highlights the value of different types of search behaviours, even if these differences are subtle.

Interestingly, in the early stages of the search, during which the steep section of the fitness landscape is descended, the exploitative operators (e.g. UNDX, SBXI and SPX) make the largest contribution to finding better solutions, as indicated by their higher *CR*(*t*) values (Figure 2-9 (a) and (d)). However, as the search progresses into the near-optimal region of the fitness landscape, the explorative operators (e.g. DCI, GMI, PM and UMI) have a bigger impact on determining better solutions (Figure 2-9 (b) and (d)). This further highlights the benefits of algorithms with a larger number of operators with diverse searching behaviours in that they have the ability to utilise the search behaviour that is most beneficial at different stages of the search.

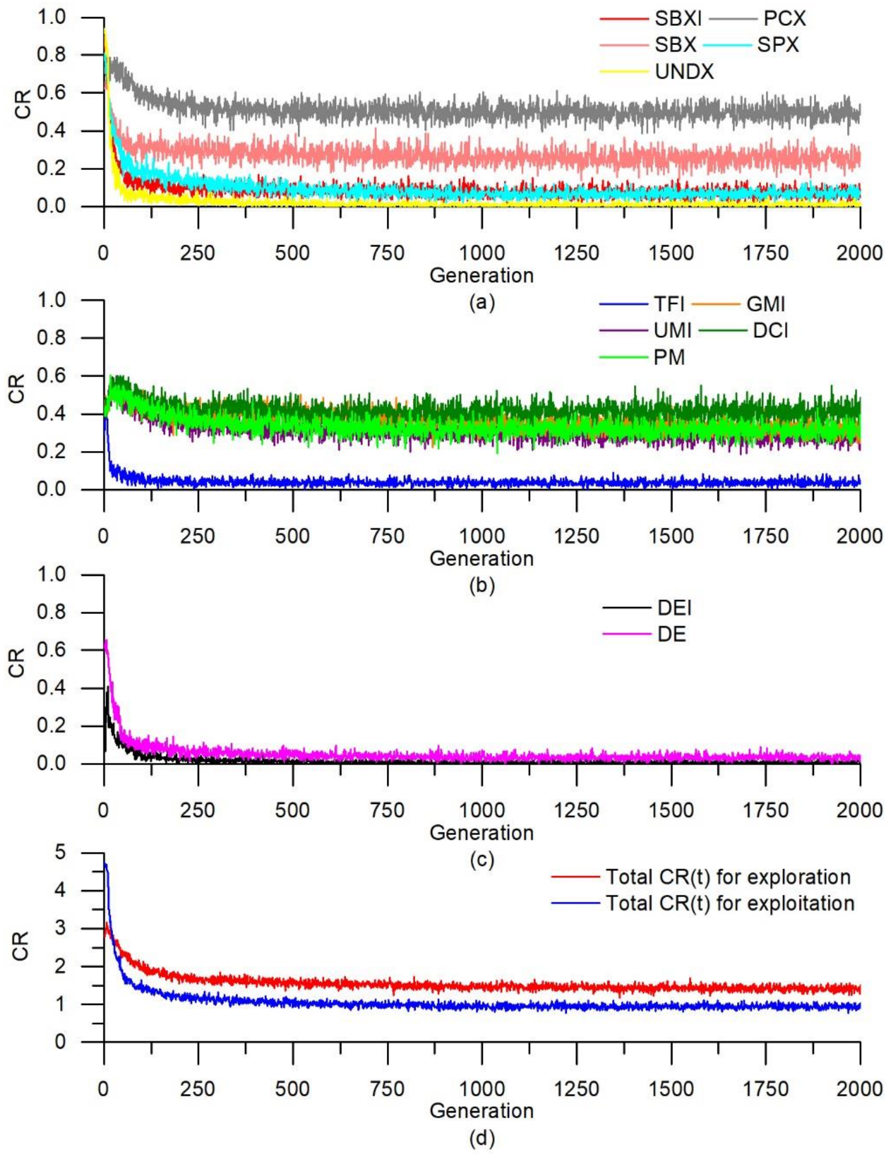


Figure 2-9. CR(t) of Algorithm 12 for the BIN problem. Note: Each line is the average value over 10 different runs. Note subfigures are: (a) for the exploitative operators; (b) for the explorative operators; (c) for the exploitative and explorative operators; and (d) total CR(t) for exploitation and exploration.

The increase in the diversity of the types of searching behaviours introduced by increasing the number of operators is the reason for the increased performance of algorithms with a larger number of operators observed in this chapter (Section 2.3.1.1). This increase in performance is also illustrated by the average values of the run-time behavioural metrics for each algorithm group (Figure 2-10). As can be seen, as the number of operators increases, so does the ability to identify solutions that (i) provide a broader range of trade-offs between the cost and resilience objectives, as evidenced by larger values of $EF(t)$ (Figure 2-10 (b)); (ii) have a lower $I_{GD}(t)$ (Figure 2-10 (c)); and (iii) have a larger $I_{HV}(t)$ (Figure 2-10 (a)). Figure 2-10 (a)-(c) also clearly show the two distinct phases of searching, with rapid improvement in the performance metrics during the descent of the steep portion of the fitness landscape in the early stages of the search, followed by a more gradual improvement as the near-optimal base of the "big bowl" in the fitness landscape is traversed.

The presence of these two distinct phases of searching is further highlighted by the changes in the population diversity metric as the search progresses (Figure 2-10), with a rapid initial decrease in solution diversity as the values of the pipe diameters that result in the largest improvements in the objectives are determined (a phase dominated by exploitation), followed by an increase in population diversity as the bottom of the "big bowl" in the fitness landscape is explored (a phase dominated by exploration). In other words, the non-dominated solutions are far from each other in decision variable space.

The two phases of searching are more distinct for algorithms with a larger number of operators. This is because the increased diversity of searching behaviours these algorithms have access to enables them to use the best possible exploitative behaviour to move down the steep portions of the fitness landscape more quickly and then the best possible explorative behaviour to navigate the relatively flat and rugged portion of the fitness landscape in the "big bowl", resulting in a larger increase in population diversity. Despite the fact that Algorithm 12 outperformed all algorithms with smaller operator sets (with statistical significance), the difference between the absolute values of the metrics (i.e. I_{HV} and I_{GD}) between Algorithm 12 and Algorithm group 10 is small, indicating that the improvement of the solution quality metrics diminishes as the operator set increases from 10 to 12 (Figure 2-10). This suggests that, in practice, an increase in the operator set size beyond 10 or 12 may not result in significant improvements in algorithm performance.

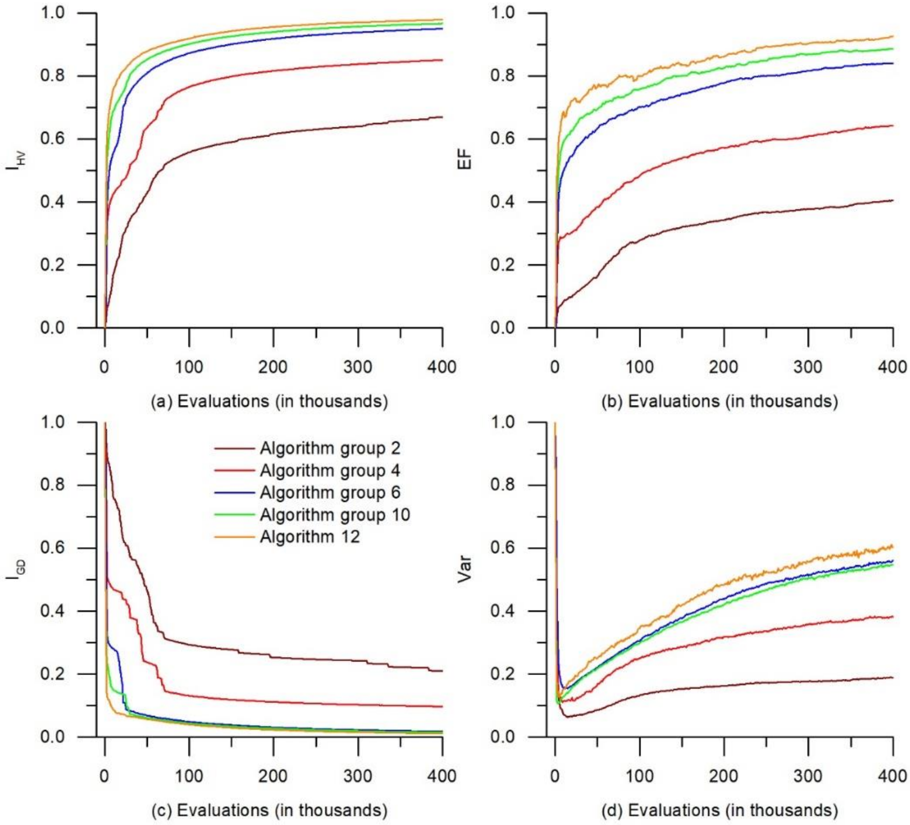


Figure 2-10. Search behaviour metrics of the different algorithm groups for the BIN problem. Note: Each line is the average value over 10 different runs across different operator sets. Note subfigures are: (a) & (c) are solution quality metrics; (b) & (d) are spacing metrics.

2.3.2 Relative Impact of the Size of the Operator Set, Operator Set Composition and Search Strategy

The rankings of the different algorithm groups with randomly generated operator sets (Experiments 1a to 1e) and those with constructed operator sets (Experiments 2a to 2d) are shown in Figure 2-11. As can be seen, the influence of the size of the operator set clearly outweighs the influence of the composition of the operator set, as indicated by the better average rank of algorithms with more operators (this is also reflected in the hypervolume values of the algorithms – this metric is considered the most comprehensive metric), irrespective of whether operator sets are generated randomly or not. While there is some variation in the ranking of algorithms with the same number of operators, this is relatively minor. Somewhat surprisingly, use of the intentionally constructed operator sets from some of the existing MOEAs resulted in a decrease in average performance compared with the use of the randomly generated operators (e.g. for constructed operator sets 2 and 3).

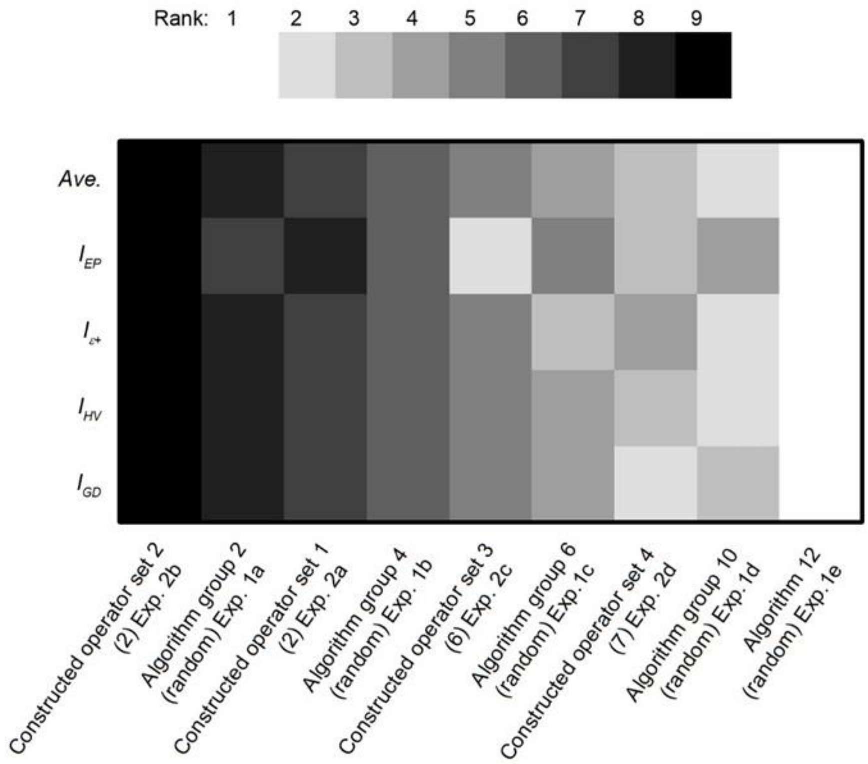


Figure 2-11. Rankings of different algorithm groups with randomly generated (Experiments 1a to 1e) and constructed operator sets (Experiments 2a to 2d)

The rankings of the different algorithm groups with randomly generated operator sets (Experiments 1a to 1e), those with constructed operator sets (Experiments 2a to 2d) and those with both constructed operator sets and search strategies (Experiments 3a to 3d) are shown in Figure 2-12. As can be seen, although the use of constructed search strategies (e.g. the use of hyperheuristics and different parent and survivor selection strategies) results in slightly larger differences in the performance of algorithms with the same number of operators, the increase in the number of operators is still the dominant factor affecting algorithm performance. This is evidenced by the clear trend of improving average rank for algorithms with a larger number of operators, irrespective of operator composition or the use of more advanced search strategies, despite some minor exceptions, including that constructed algorithm 1 (2 operators) performs slightly better than Algorithm group 4 and constructed algorithm 4 (7 operators) performs worse than Algorithm group 4 and constructed algorithm 1 (2 operators).

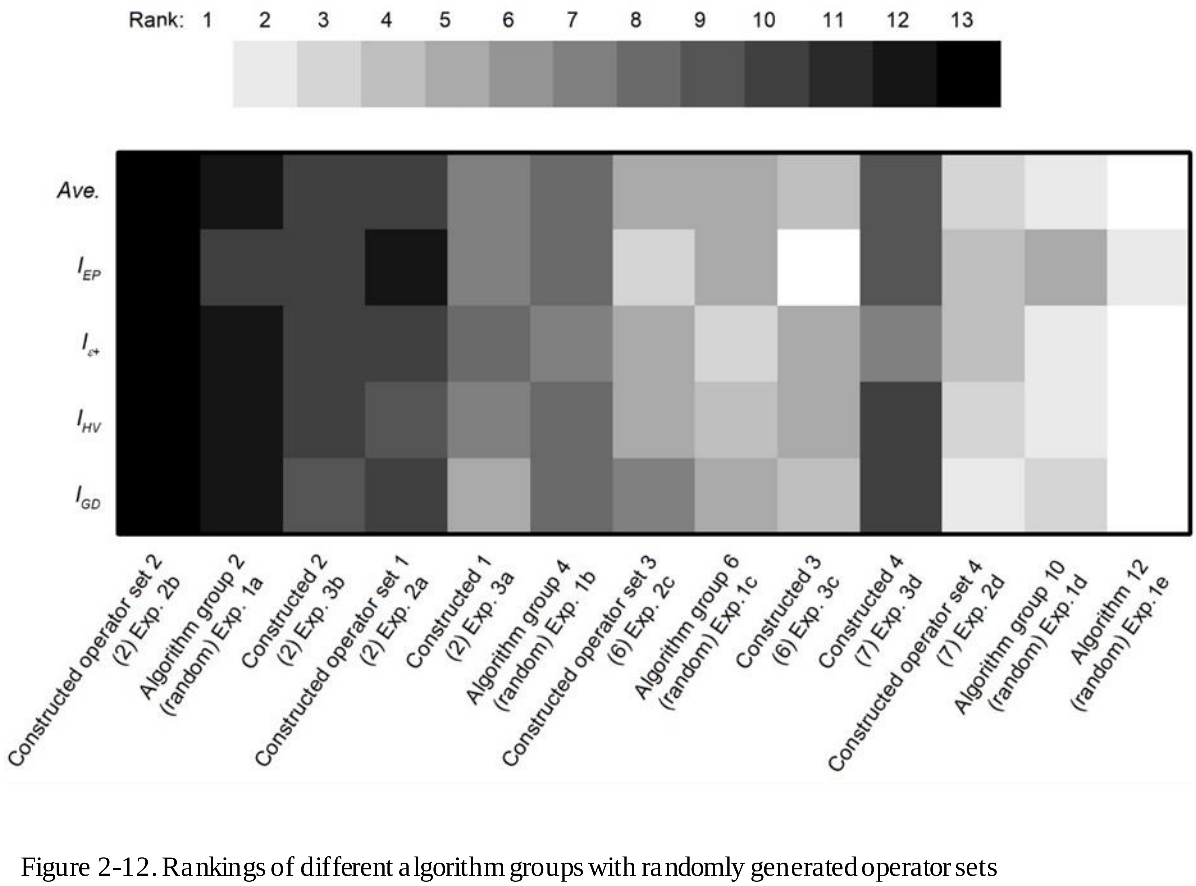


Figure 2-12. Rankings of different algorithm groups with randomly generated operator sets (Experiments 1a to 1e), constructed operator sets (Experiments 2a to 2d) and constructed operator sets and search strategies (Experiments 3a to 3d)

2.3.3 Potential for Improving the Performance of Existing Algorithms by Increasing their Operator Set Size

As GALAXY performed best out of the four algorithms considered (i.e. the algorithm used in Experiment 3c performed better than the algorithms used in Experiments 3a, 3b and 3d - Figure 2-12), its operator set was expanded to include all 12 operators considered in this paper (GALAXY-12 - Experiment 4, Figure 2). For benchmarking purpose, the performance of this algorithm was compared with that of the original GALAXY (6 operators – Experiment 2c) and Algorithm 12 (Experiment 1e), which was the best performing algorithm in all previous experiments (see Figure 2-11 and Figure 2-12). The results show that by expanding the size of the operator set of GALAXY from 6 to 12, its performance can be improved significantly, to the point where its performance is better than that of any other algorithm tested in this chapter (Figure 2-13). The fact that GALAXY-12 (12 operators) outperforms Algorithm 12 demonstrates that the advanced features of GALAXY (e.g. hyperheuristics, alternative parent and survivor selection strategies) are able to improve algorithm performance. These results suggest that there is potential of improving the performance of existing MOEAs by increasing the size of their operator sets.

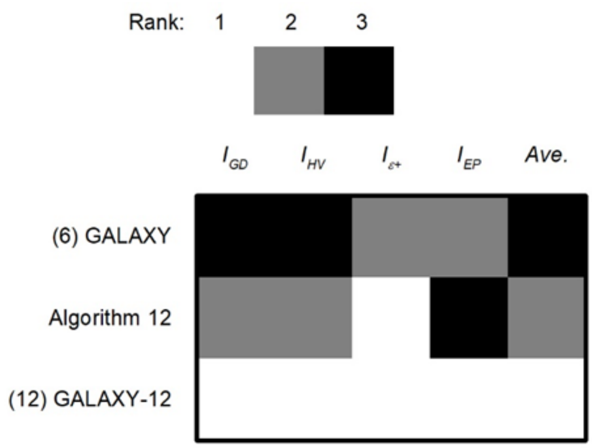


Figure 2-13. Ranking of GALAXY (6 operators – Experiment 2c), GALAXY with an expanded set of 12 operators (GALAXY-12 – Experiment 4) and Algorithm 12 (Experiment 1e)

2.4 Conclusion

This paper has studied the impact of operator set size on the performance of multi-objective evolutionary algorithms (MOEAs) for six WDS problems. Specifically, the research objectives were to assess (i) the relative influence of the size of the operator set on algorithm performance, (ii) whether the size of the operator set is more important than the composition of the operator set, (iii) whether the size of the operator set is more important than the combined effect of the composition of the operator set and the search strategies used, and (iv) the potential for improving the performance of existing MOEAs by increasing the size of the operator set.

In order to isolate the influence of operator set size and composition from that of other search strategies, a generic MOEA framework was developed. The influence of operator set size was assessed by randomly selecting operators from a pool of 12 (i.e. 6 explorative and 6 exploitative) for operator set sizes of 2, 4, 6, 10 and 12 and the influence of operator set composition was assessed by using the constructed operator sets from four well-known

MOEAs, instead of randomly selected operators. The impact of other searching strategies was assessed by using the four unique constructed search strategies from the four well-known MOEAs (not just their operators) in lieu of the MOEA framework and the potential of improving the performance of existing MOEAs by increasing their number of operators was assessed by including all 12 operators considered in this chapter in the well-known MOEA that performed best for the case studies considered, which was GALAXY.

The results from the 3,150 optimisation runs clearly indicate that operator set size plays a dominant role in algorithm performance for the six WDS case studies considered. Operator set size had a larger influence than operator parameter values, operator set composition and other strategies affecting algorithm searching behaviour. The reason for the increased performance of algorithms using a larger number of operators is that they provide a larger variety of searching mechanisms, which are able to find better solutions at different stages of the optimisation process.

Given the complexity of the problems considered in this chapter, the general finding that algorithm performance can be improved by increasing the size of the operator set used should hold for a wider class of combinational optimisation problems. However, it is important to note that specific conclusions with regard to the relative performance of particular algorithms are conditioned on the WDS design problems, which are possible to have a “big bowl” shape in their fitness landscape. Its generalization to other problem types need further investigations, with focus on the assessment of the controllability, effectiveness, efficiency and reliability (Hadka & Reed, 2012). For example, given that GALAXY was specifically tailored for the optimisation of WDS problems (in fact, GALAXY can only be applied to problems with discrete decision variables), it is less likely to perform as well as more generic algorithms, such as Borg, which has shown consistent levels of controllability, effectiveness, efficiency, and reliability on different multimodal and non-separable problems (Hadka & Reed, 2012). Consequently, future work should extend the assessment of the impact of operator set size on algorithm performance to a broader array of problem types, where issues of scalability across dimensions and objectives can be considered directly, as in Kollat and Reed (2007).

Overall, the findings of this chapter tend to suggest that existing multi-objective evolutionary algorithms do not use a sufficient number of operators and that there is significant potential to increase the performance of a wide range of existing algorithms by simply increasing their operator set size. Based on the results obtained, it is recommended to increase the number of operators in existing algorithms to 10 or 12, ensuring a balance between exploration and exploitation (see Table 2-1 for guidance). For cases where the original algorithm to be improved does not use a hyperheuristic to control the degree to which each operator contributes to the search at each iteration, it is recommended to use the NAÏVE hyperheuristic, which ensures that all operators contribute equally.

Chapter 3 A New Proposed Convex Hull Selection Strategy: Study on the Impact of Selection Strategy on Generational MOEA Performance for Water Distribution Design Problems

Peng Wang[1,2], Aaron C. Zecchin[1], Holger R. Maier[1,2]

[1]School of Civil, Environmental and Mining Engineering, The University of Adelaide, Adelaide, SA, Australia;

[2]College of Civil Engineering and Architecture, Zhejiang University, China

Statement of Authorship

Title of Paper	A New Proposed Convex Hull Selection Strategy: Study on the Impact of Selection Strategy on Generational MOEA Performance for Water Distribution Design Problems	
Publication Status	☐ Published ☐ Submitted for Publication	☐ Accepted for Publication ☑ Unpublished and Unsubmitted work written in manuscript style
Publication Details		

Principal Author

Name of Principal Author (Candidate)	Peng Wang		
Contribution to the Paper	Primary innovator, analyst and author Conception and design of the project Development and execution of numerical experimental program Analysis and interpretation of research data Draft the paper		
Overall percentage (%)	90		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	01/10/2020

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);
ii. permission is granted for the candidate in include the publication in the thesis; and
iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Aaron C. Zecchin		
Contribution to the Paper	Conception and design of the project Analysis and interpretation of research data Critically revising the paper so as to contribute to the interpretation		
Signature		Date	17/07/2020

Name of Co-Author	Holger R. Maier		
Contribution to the Paper	Conception and design of the project Analysis and interpretation of research data Critically revising the paper so as to contribute to the interpretation		
Signature		Date	09/09/2020

Abstract

Multi-objective evolutionary algorithms (MOEAs) have been applied to water distribution system (WDS) optimisation problems for over 20 decades. The selection strategy is a key component of a MOEA that determines the composition of a population, and thereby the evolutionary search process, which imitates natural selection by granting fitter individuals an increasing opportunity to reproduce (Yu and Gen, 2010). This paper proposes a novel selection strategy for generational MOEAs that is based on the convex hull contribution of solutions to the Pareto front in the objective space. Numerical experiments using a general MOEA framework, demonstrate that the proposed selection strategy is able to outperform existing popular selection strategies (e.g. crowding distance, Hypervolume contribution, and hybrid replacement selection strategies). Moreover, it is illustrated that the CHC_{Gen} selection strategy is able to improve the performance of existing MOEAs. The conclusions are based on the results of six bi-objective WDS problems.

Keywords: Multi-objective evolutionary algorithms, selection strategy, water distribution system design optimization, GALAXY, NSGA-II.

3.1. Introduction

Water resource optimisation problems are commonly characterised by non-linearity, multi-modality, high variable interdependencies, search space discontinuities, high variable dimensionality and multiple objectives (Nicklow et al., 2010; Szemis et al., 2013; Maier et al., 2019). In the last thirty years, the use of multi-objective evolutionary algorithms (MOEAs), and other metaheuristics, has arguably become the preferred optimisation approach in water resources (e.g. Maier et al., 2014; Mala-Jetmarova et al., 2017, 2018). The MOEA process can be viewed as the selection and perturbation of an existing population of solutions by sequential application of specific operators (e.g. crossover and mutation), with the aim of generating new and improved solutions to iteratively refine the quality of solutions in the population set (Asadzadeha et al., 2014).

A key advantage of MOEAs is the identification of a set of solutions that provide near optimal trade-offs between competing objectives in a single optimisation run (Deb et al., 2002b). These solutions are called the approximate Pareto optimal solutions and the set of these solutions is called the approximate Pareto front (as they provide an approximation to the true Pareto front). Typically, the shape of approximate front (e.g. whether it is convex or nonconvex) is unknown a priori and depends on a problem's characteristics. However, the shape of the approximate front is found to be convex for many practical problems in practice. For example, Asadzadeha et al., (2014) stated that for multi-objective hydrologic model calibration, the approximate front shape is usually convex (i.e. a knee-bend in the middle and long tails). This statement is supported by numerical experiments of hydrological modelling calibration (Xia et al., 2002; Fenicia et al., 2007; Lee et al., 2011 & Kollat et al., 2012). For water distribution system (WDS) problems, Wang et al., (2015) evaluated 12 WDS case studies, within this work, the approximate front shapes were all found to be convex.

As the actual Pareto front is not known for most real-life problems, the performance of different MOEAs' ability to generate a quality approximate front is compared using metrics that correspond to different attributes of the front. For example, the average distance of the approximate front to the "ideal point" (i.e. the point in the objective space that dominates all other points) is compared to understand MOEA's ability of convergence, where the front closest to this point is preferred. Alternatively, the extent of the front is measured by evaluating the diversity of the set of solutions (e.g. the degree of spreading).

The ability of an MOEA to generate a quality approximate front is affected by a number of search strategies including the following factors: the operators govern how offspring solutions are produced; the hyperheuristic manages the utilisation of each operator throughout the search; and the selection strategy determines which solutions are selected to be used by the operators to produce the offspring solutions, and which offspring solutions survive to join the population. Based on how the selection strategy is utilised, there are two types of MOEAs, namely, generational and steady-state. A generational MOEA generate multiple offspring solutions within each generation, where generally the number of offspring solutions is equal to the size of population. As a consequence of this process, the offspring solutions compete between themselves and the existing population to enable survival of the best solutions (Zapotecas & Menchaca, 2020). A steady-state MOEA involves the selection of two individuals from the population to generate a singular offspring within each generation. The new solution replaces the worst performing solution from the population. Within both of these algorithms, the selection strategy is particularly important as it needs to be designed to drive the population to converge to increasingly fit regions of the search space, whilst avoiding premature convergence to sub-optimal regions (Back, 1996; Hanne, 1999).

In the past 20 years, many selection strategies have been proposed and applied within both types of MOEAs. For steady-state MOEAs, several studies have investigated the impact of different selection strategies on algorithm performance. For example, Emmerich et al., (2005) applied a hypervolume contribution (HVC1) selection strategy (Knowles et al., 2003) to a steady-state multi-objective selection based on dominated hypervolume (SMS-EMOEA). The HVC1 captures both attributes of a Pareto Front in terms of convergence and diversity by measuring the contribution of to the overall hypervolume (Zitzler & Thiele, 1999) of a Pareto front by each individual solution. In the comparative experiments, HVC1 outperformed NSGA-II (a generational algorithm) with a crowding distance (CD) selection strategy, which focuses on maintaining Pareto front diversity (Deb et al., 2002b). However, these studies did not isolate the performance impact of the selection strategy from that of the other algorithm processes, which poses a difficulty in terms of being able to attribute any performance differences to HVC1. This limitation was addressed by Igel & Hansen, (2007) who compared the two selection strategies (HVC1 and CD) by applying them within the same generational algorithm, the multi-objective covariance matrix adaptation evolution strategy (MO-CMA-ES). The results suggested the algorithm with HVC1 outperformed those with a CD selection strategy on a range of test functions. Building on this work, Asadzadeh & Tolson (2013) compared the influence of four selection strategies using a steady-state MOEA termed Pareto archived dynamically dimensioned search (PA-DDS). The selection strategies considered in this work were HVC1, an alternative Hypervolume contribution termed HVC2 based on the work of Bader & Zitzler (2001), CD and a purely random strategy (RND) which did not take account of any of the solutions attributes. The mechanism of HVC2 is akin to HVC1 but has an additional parameter that defines the maximum number of solutions that should be considered in the calculation of HVC2 [the interested reader should refer to (Bader & Zitzler 2001)]. The comparative study showed that the HVC1 selection strategy achieved the best performance overall in all test functions and water resources problems. Moreover, in the water resources problems considered in Asadzadeh & Tolson (2013), the HVC1 selection strategy resulted in better solutions found within the 'knee region' of the approximate fronts. The reason for this is that more solutions in the 'knee region' are improved during the search, as the solutions in this region normally have a greater hypervolume contribution value, thereby having more chance to be selected to produce solutions, and be retained in the population (Asadzadeh & Tolson, 2013; Jahanpour et al., 2018). In practice, the solutions in the 'knee region' of a Pareto front are preferred as they provide a locally distinct compromise of each objective (Mala-Jetmarova et al., 2018; Hadka & Reed, 2012).

For a convex Pareto front, Feng et al., (1997) and Cococcioni et al., (2007) showed that giving greater selection priority to solutions that are closer to the convex approximation of the Pareto front can improve the performance of an MOEA. Asadzadeh et al., (2014) proposed the novel convex-hull contribution (CHC) selection strategy for PA-DDS. This approach gives a high selection priority to the non-dominated solutions that have greater CHC values. Typically, a solution CHC is the difference in size (e.g. area) of the convex hull set between the approximate front with and without that solution. This selection strategy not only accounts for convergence and diversity in generating the approximate front, but its search behaviour is well suited for problems that have a convex shape of the approximate front, such as the WDS problem. Moreover, given the nature of the convex shape of approximate fronts, the solutions in the 'knee region' have a greater CHC value thereby being selected and exploited. Thus, the CHC selection strategy is effective on finding better solutions in the 'knee region' of the approximate fronts (Asadzadeha et al., 2014 and Jahanpour et al., 2018). This selection strategy is currently the top one for water resource problems, as it has been found to improve the performance of PA-DDS, and outperformed the HVC1 selection strategy in test functions, real hydrologic model calibration problems and WDS problems (Asadzadeha et al., 2014 and Jahanpour et al., 2018). In particular, this work found that the CHC selection strategy resulted in better solutions being identified within the ‘knee region’ of the approximate fronts, in comparison to HVC1. However, there is currently no generic formulation of CHC selection strategy that is applicable to generational MOEAs.

Consequently, the objectives of this chapter are to: (i) develop a new convex hull selection strategy formulation for generational MOEAs (termed CHCGen); (ii) explore the impact of different selection strategies for generational MOEAs; and (iii) test the utility of the CHCGen selection strategy to improve existing MOEA’s performance. The above objectives are achieved by conducting an extensive numerical experimental program involving WDS case studies as in the work of Wang et al., (2015, 2017); Zheng et al., (2016, 2017).

The remainder of this paper is structured as follows. The section Methodology formulates the CHCGen selection strategy, and a general MOEA framework is introduced to provide a generic generational algorithm structure to investigate the impact of different selection strategies. Additionally, two existing MOEAs are outlined with modifications proposed by including the CHCGen to evaluate its utility for other generational MOEAs. The structure of the computational experiments is also outlined. In the section Result and Discussion, the results of the numerical experiments were reported and highlighted the effectiveness of CHC*Gen* selection strategy. The summary of the finding was concluded at the end.

3.2. Methodology

The general steps within a multi-objective evolutionary algorithm (MOEA) is shown in Figure 3-1. As can be seen, at the commencement of the optimisation process, an initial set of solutions is randomly generated and form the population $\boldsymbol{x_t}$ (t=0) with size N. Then, subject to the parent selection process, some solutions are selected as parent solutions that have opportunity to reproduce offspring $\boldsymbol{y_t}$. The reproduction process is facilitated by one or more operators. In addition, the degree to which operator contributes to the search at each generation can be controlled with the aid of hyperheuristics, which are high-level automated search strategies for selecting the most appropriate lower-level operators (or heuristics, such as mutation and cross-over) (Burke et al., 2013; Drake et al., 2019). Thereafter, $\boldsymbol{x_t}$ and $\boldsymbol{y_t}$ are combined to form a combined set $\boldsymbol{c_t}$. The replacement is carried out to select and to form the next generation $\boldsymbol{x_{t+1}}$. The above process is repeated until certain termination criteria are met, such as the execution of a fixed number of generations or no better solutions are identified.

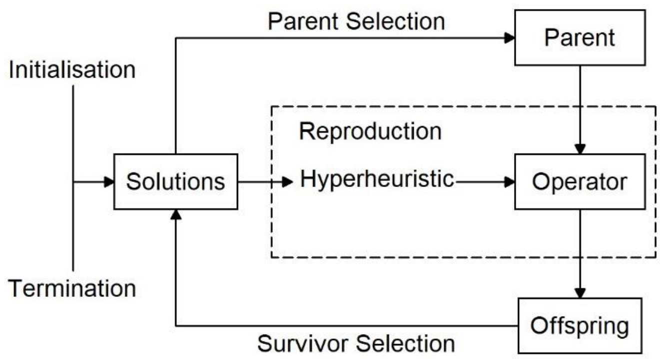


Figure 3-1. Generic multi-objective evolutionary optimisation process

The purpose of the process of parent selection and replacement is to identify better solutions from the current solution set. A general process of parent selection and replacement is summarised as follows. For parent selection, in order to select better solutions from population set $\boldsymbol{x}$, the primary criterion value for solution x_i is given by $l(x_i|\boldsymbol{x})$, where l indicates the relative rank of x_i with respect to the population $\boldsymbol{x}$ (e.g. non-dominance rank). Given this criteria value assignment, the population is organised into the sets $\boldsymbol{\mu}_1, \ldots, \boldsymbol{\mu}_g$, based on their values, where if x_i and x_j are in $\boldsymbol{\mu}_K$, then $l(x_i|\boldsymbol{x}) = l(x_j|\boldsymbol{x})$, and $l(x_i|\boldsymbol{x}) < l(x_k|\boldsymbol{x})$ for all $x_k \in \boldsymbol{\mu}_G$ for which $G > K$ (i.e. each $\boldsymbol{\mu}_K$ contains solutions of the same criteria value, and indicate a better rank for lower K). The secondary selection criterion applies to each solution in each $\boldsymbol{\mu}_K$ and is denoted as $v(x|\boldsymbol{\mu}_K)$, $x \in \boldsymbol{\mu}_K$ which enables the ordering of solutions in $\boldsymbol{\mu}_K$ as $\left(x_{[1]}, x_{[2]}, \ldots, x_{[|\boldsymbol{\mu}_K|]}\right)$ where $v(x_{[i]}|\boldsymbol{\mu}_K) > v(x_{[j]}|\boldsymbol{\mu}_K)$ if $j > i$. The sorting of the population $\boldsymbol{x}$ is then based on the overall rank of a solution with respect to the entire population, which is first based on I (i.e. the $\boldsymbol{\mu}_K$ set that a solution is a member of) and secondarily on v. As indicated, in general, the primary selection criterion is associated with solution quality (e.g. nondominance rank) and the secondary selection criterion is associated with the diversity of solutions (e.g. crowding distance).

The replacement follows a similar approach as with the parent selection that is outlined in Figure 3-2. This stage involves the sorting of $\boldsymbol{c}_t$ to select N solutions to form $\boldsymbol{x}_{t+1}$. First $\boldsymbol{c}_t$ undergoes a non-dominance sorting process which involves the allocation of solutions in $\boldsymbol{c}_t$ into the ordered subsets $\boldsymbol{\mu}_1, \boldsymbol{\mu}_2 \ldots,$ where $\boldsymbol{\mu}_i$ indicates the solutions from $\boldsymbol{c}_t$ that form the i-th order Pareto front (e.g. $\boldsymbol{\mu}_1$ are the pareto optimal solutions for the entire set $\boldsymbol{c}_t$, $\boldsymbol{\mu}_2$ are the pareto optimal solutions for the reduced set $\boldsymbol{c}_t \backslash \boldsymbol{\mu}_1$ and so on). As $|\boldsymbol{\mu}_1|$ is typically smaller than N, all of the solutions in $\boldsymbol{\mu}_1$ are used to form the first $|\boldsymbol{\mu}_1|$ solutions in $\boldsymbol{x}_{t+1}$. The remaining available $N - |\boldsymbol{\mu}_1|$ positions in $\boldsymbol{x}_{t+1}$ are filled by subsequent solutions in $\boldsymbol{\mu}_2$. This process is repeated until the point K in which $|\boldsymbol{\mu}_1| + \cdots + |\boldsymbol{\mu}_{K+1}| > N$. At this point, the set $\boldsymbol{\mu}_{K+1}$ is sorted according to the second selection criteria, where the top ranked solutions are used to fill the remaining positions in $\boldsymbol{x}_{t+1}$. This step represents the final step for a given generation.

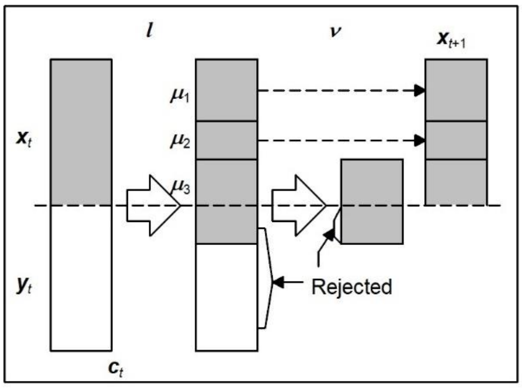


Figure 3-2. Replacement step for a generational MOEA. (Deb et al., 2002b)

3.2.1. Proposed new CHC Selection Strategy

In order to develop a selection strategy for generational MOEA that not only accounts for convergence and diversity in generating the approximate fronts, but its search behaviour is well suited for problems that have convex shape of approximate front. Moreover, the selection strategy is desired to result in better solutions being identified within the 'knee region' of approximate Pareto fronts. The CHC selection strategy for generational MOEA (termed as CHC_{Gen}) is introduced in this section. Initially, the concepts of convex hull and convex hull contribution are outlined. The approach of calculating CHC value for a solution is based on Asadzadeh et al., (2014) for either generational or steady state MOEAs. Thereafter, the detail procedure of the CHC_{Gen} is introduced.

3.2.1.1. Convex hull background

For an m-dimensional space, the convex hull of a set of points $S \in \mathbb{R}^m$ is the intersection of all convex sets containing S (Barber et al., 1996). An example of the convex hull (grey filled area) of a given set of points (empty circles) set in a 2-dimensional space is given in Figure 3-3. It can be seen that the convex hull area contains all of the given points. From Figure 3-3, a convex hull is bound by the segment line of two vertices (dot points) called facets (solid and dash lines). The convex hull contribution is described as follows.

Each facet divides the space into two sides, one side contains the convex hull and the other does not. The area of the region bounded by the facets (or area in a 2-dimensional space) is called the convex hull size (V). If any of the vertices (s) is removed from a convex hull set, the size of the convex hull will change (Asadzadeh et al., 2014). For instance, if the vertex *p* is removed from the convex hull in Figure 3-3, the new facet (the dot dash line) is the new boundary of the new convex hull, thereby reducing the size of the convex hull by an amount given by the sparse lined area in Figure 3-3. In *m*-dimensional space, the reduced volume (or area) that is resulted by removing any vertexes (e.g. that given by the lined area) is called a convex hull contribution (CHC), and is given as

$$\mathrm{CHC} = V(\mathbf{S}) - V(\mathbf{S} \setminus p) \quad (3\text{-}1)$$

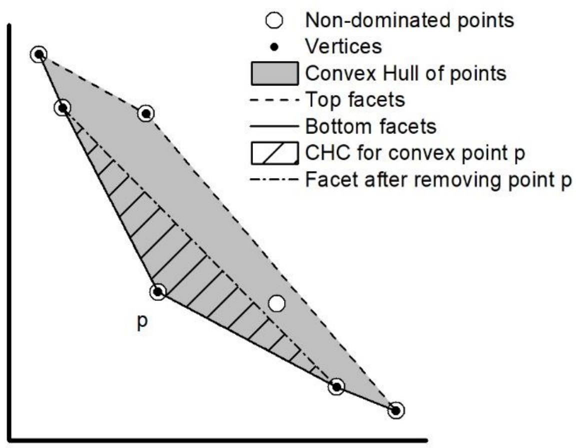


Figure 3-3. Convex hull of nondominated points in a bi-objective space. (Asadzadeh et al., 2014)

The key steps for CHC calculation for a nondominated set is outlined as follows (interested readers should refer to Asadzadeh et al., (2014) for more details). For the first step, the axes of a nondominated set are normalized in the objective space to eliminate any potential bias resulted by varying scales of the objective functions. For the following explanation, consider that the 2-D space in Figure 3-3 is representative of an objective space, where the criteria for both objectives is a minimisation (that is, the lower left side of the polygon is associated with the Pareto front). The solutions are divided into four groups as shown in Figure 3-3: (i) points inside the convex hull; (ii) vertices on the top facets (dash line); (iii) vertices on the bottom facets (solid line) and (iv) vertices in the intersection of top and bottom facets.

To calculate the CHC for each solution, zero CHC values are assigned to the solutions in group (i) and (ii) that are mentioned above. The calculation of CHC is only conducted for solutions in group (iii), where, for example, the CHC for point *p* is given by the sparse lined area in Figure 3-3. The CHC for the solutions in group (iv) is assigned as the CHC value of the closest solutions' CHC values from group (iii) (Asadzadeh et al., 2014). The original CHC selection strategy is designed for a steady state MOEA (the aforementioned PA-DDS), for which the population set only contains non-dominated solutions. However, for a generational MOEA, the CHC selection strategy is not applicable for two reasons. Firstly, a generational MOEA's population set may contain dominated solutions during the search (Deb et al., 2002b), which are beneficial for solution diversity. In this case, dominated solutions may have greater CHC values than nondominated solutions. For example, as outlined in Davoodi et al., (2011), the solution at the top facet is dominated, but has a positive CHC value. Applying the CHC selection strategy directly within a generational MOEA would result in a higher selection probability to such dominated solutions, thereby potentially resulting in a poor performance. Secondly, the portion of solutions with a non-zero convex hull contribution from a given set is usually small (Jahanpour et al., 2018). In other words, a great portion of solutions' CHC values are zero and, as a result, they become non-competitive in a tournament selection process. This will result in a highly biased selection process and may not benefit an algorithm's performance. In summary, there is a motivation to propose a new CHC selection strategy for generational MOEAs.

3.2.1.2. Proposed new CHC for a population based MOEA

The details of a proposed new generational CHC (CHC_{GEN}) selection strategy for population based MOEAs is outlined in Figure 3-4. At the beginning, the fast non-dominance sorting approach is implemented to the combined set $\boldsymbol{c}_t$ to sort the solutions into the non-dominance sets $\boldsymbol{ND}_1, \boldsymbol{ND}_2, \ldots$ (line 1), where $\boldsymbol{ND}_i$, is the i-th non-dominant front, where every solution in $\boldsymbol{ND}_i$ domainates all solutions in $\boldsymbol{ND}_j$, for $j > i$. Within the proposed approach, the primary selection criterion is the non-dominance rank and the secondary selection criterion is the CHC value. The values for $\boldsymbol{\nu}$ for each solution are initialised as Ø; convex hull contribution to $\boldsymbol{ND}_q$, CHC($\boldsymbol{ND}_q$) is initialised as 0; nondominance convex hull sets $\boldsymbol{\mu}$ is initialised as Ø (line 2). Then, the process evaluates the convex hull contribution to the solutions in each $\boldsymbol{ND}_q$ set (line 5) and removes the solutions with positive convex hull contribution values from $\boldsymbol{ND}_q$ and inserts them into $\boldsymbol{\mu}_l$. The $\boldsymbol{\nu}^l$ of the solutions in set $\boldsymbol{\mu}_l$ is updated as well.. The above procedures are iterated until the set $\boldsymbol{ND}$ is emptied. At this stage, all of the solutions are categorised into $\boldsymbol{\mu}$ and associated by convex hull rank l and convex hull contribution $\boldsymbol{\nu}$. Consequently, the replacement step is enable to be conducted as mentioned above. The $\boldsymbol{x}_{t+1}$ is filled by N solutions from $\boldsymbol{c}_t$, where the solutions with smaller l and greater ν values are preferred.

Inputs: Input combined set $\boldsymbol{c}_t$ that requires selection of N solutions ($N < |\boldsymbol{c}_t|$).
Implement fast nondominated sorting approach to find the nondominated fronts $\boldsymbol{ND} = (\boldsymbol{ND}_1, \ldots, \boldsymbol{ND}_r)$ of $\boldsymbol{c}_t$
Set: $\boldsymbol{x}_{t+1} = \emptyset$, $l = 1$, $\boldsymbol{\mu} = \emptyset$, $\boldsymbol{\nu} = \emptyset$, $q = 1$, CHC($\boldsymbol{ND}$) = 0
While $q \leq r$
    for all $\boldsymbol{ND}_q \neq \emptyset$
        Evaluate CHC($\boldsymbol{ND}_q$)
        $\boldsymbol{temp} = \{ \boldsymbol{ND}_q$, CHC($\boldsymbol{ND}_q$)>0$\}$
        $\boldsymbol{ND}_q = \boldsymbol{ND}_q \setminus \boldsymbol{temp}$
        Add $\boldsymbol{temp}$ to $\boldsymbol{\mu}_l$
        $\boldsymbol{\nu}^l =$ CHC($\boldsymbol{ND}_q$)>0
        $l = l + 1$
    $q = q + 1$

Figure 3-4. Proposed $\mathrm{CHC_{Gen}}$ selection strategy

3.2.2. Numerical Experiment

In order to study the impact of the proposed new $\mathrm{CHC_{Gen}}$ selection strategy on MOEA performance, a systematic approach has been adopted to compare the proposed new $\mathrm{CHC_{Gen}}$ selection strategy with other existing selection strategies (Objective 2); and to investigate the application of the new $\mathrm{CHC_{Gen}}$ selection strategy on existing MOEAs (Objective 3). The flow chart of the numerical experimental program is proposed and shown in Figure 3-5.

In order to investigate the impact of different selection strategies on a MOEA performance (Objective 2, Figure 3-5), it is important to isolate the impact of selection strategy. The general MOEA framework (Wang et al., 2020a) is a generational MOEA that has interchangeable components (e.g. operators, hyperheuristic and selection strategy) and is adopted to achieve this target. As part of this framework, the operator set and hyperheuristic remain unchanged, and the selection strategies of parent selection and replacement are varied. Each constructed MOEA is named by the selection strategy utilised. For example, the general MEOA framework utilising the crowding distance selection strategy is denoted as Algorithm-CD. In addition, to measure the absolute performance of the general MOEA framework with the proposed new $\mathrm{CHC_{Gen}}$ selection strategy, two popular generation MOEAs applied to water resources problems, NSGA-II and GALAXY, were modified by embedding the new CHC_{Gen} selection strategy and comparing with the original version of the algorithms.

In order to assess the influence of different selection strategies on MOEA performance, the proposed new CHC_{Gen} and four existing selection strategies, which are hypervolume contribution (HVC), crowding distance (CD), hybrid replacement (HR), random (RND) were adopted within the general MOEA framework and compared with each other (i.e. in the largest solid block in Figure 3-5).

In order to test the utility of the proposed new CHC_{Gen} selection strategy to improve existing MOEAs. The proposed new CHC_{Gen} selection strategy was adopted within the two existing MOEAs that are denoted as NSGA-II-CHC_{Gen} and GALAXY-CHC_{Gen}. The two modified algorithms were compared with the original versions of these algorithms. Moreover, the general MOEA framework with the proposed new CHC_{Gen} strategy was included in the comparison in order to investigate the relative influence of other components such as operator and hyperheuristic on MOEA performance.

The above numerical experiments were assessed by using a bi-objective optimization problem (i.e. minimising network cost and maximising network resilience) for the six WDS problems shown in Figure 3-5. These case studies include: the New York tunnel network (NYT), the Hanoi network (HAN), the Fossolo network (FOS), the Pescara network (PES), the Modena network (MOD) and the Balerma irrigation network (BIN). These cases studies have been widely used to assess MOEAs performance (Wang et al., 2015, 2017 and Zheng et al., 2016). As shown in the block at the central of Figure 3-5, each optimization runs were duplicated 10 times with different starting positions in decision domain.

In the "Result Assessment" block at the bottom of Figure 3-5, the results for the different objectives are compared by applying the one-way Kruskal-Wallis test (Kruskal & Wallis, 1952) with Dunn's D post-test (Dunn, 1964) to three end-of-run performance metrics. Moreover, a novel visualization metric called the selection metric is introduced and implemented to understand how different search strategies affect an algorithm's search.

The code of the general MOEA framework was adopted from Wang et al., (2020); the code of the selection strategies considered in this chapter were written in MATLAB *m*-script; and the CHC_{Gen} selection strategy, the convex hull's size (i.e. Lebesgue measure), vertices, facets, etc. can be acquired by implementing the "qhull" code, which is available at: http://www.qhull.org/ (Barber et al.,et al., 1996).The best-known solutions were compared with reference Pareto fronts (the best-known Pareto fronts) for the six WDS case studies found by Wang et al., (2015) and Jahanpour et al., (2018). All simulations were run on the Phoenix High Performance Computer (HPC) at the University of Adelaide, Australia. The Phoenix HPC is a heterogeneous hardware system that includes a mix of CPU-only and CPU/GPU-accelerated nodes. It has 260 nodes in total, which are equipped with 2x Intel Gold 6148, 40 cores @ 2.4GHz, and 384GB memory for CPU nodes. In addition, the maximum RAM per node is 125 GB.

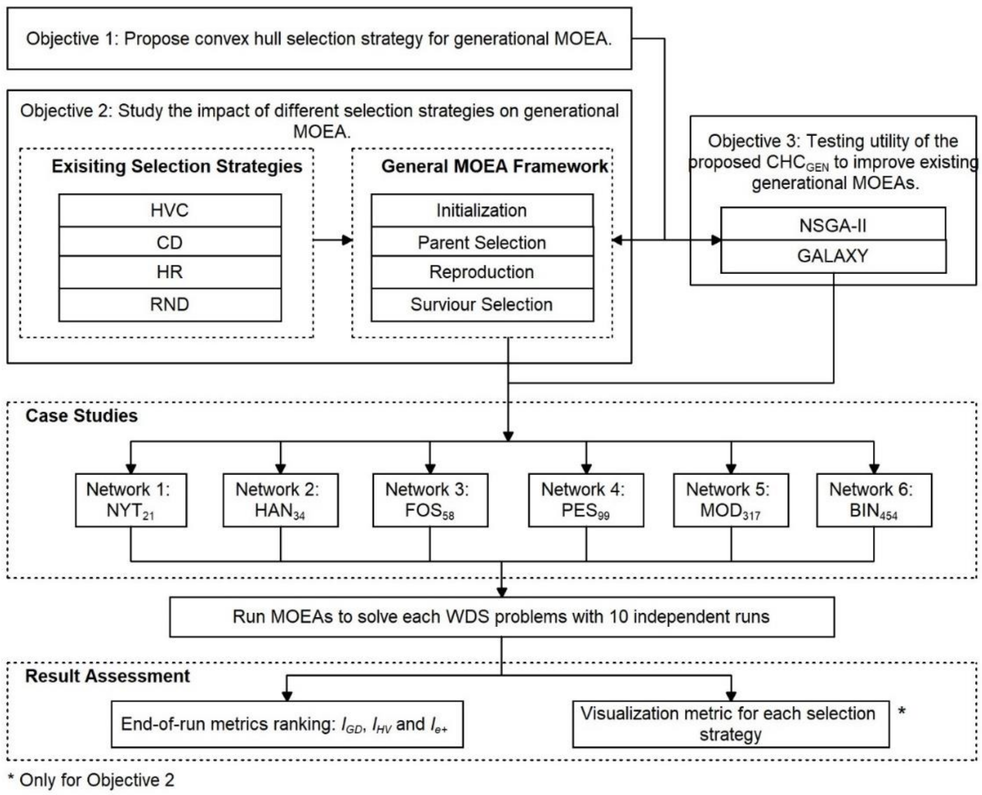


Figure 3-5. Overview of the methodology flowchart for each objective.

3.2.3. Comparison Selection Strategies

Four existing selection strategies, studied for the purpose of comparison, are described in this section. The reason for considering the four existing selection strategies in this chapter are that they have been implemented by popular generational MOEAs (e.g. NSGA-II, GALAXY) and have shown effective performance in WDS problems (Wang et al., 2015 & 2017). Each selection strategy is described briefly by the selection criteria outlined in Section 3.2.1. For example, for the crowding distance selection strategy, the primary selection criterion l is defined as the nondominace rank; and the secondary selection criterion ν is defined as crowding distance. Interested readers can find details to each selection strategy in the corresponded references outlined in the following subsections.

3.2.3.1.Crowding Distance

The crowding distance (CD) selection strategy is a popular selection strategy that has been widely applied in many MOEAs since it was proposed by Deb et al., (2002b). It uses nondominace ranks as l, and CD as v. CD measures the diversity of solutions within the same nondominated front in the objective space. In the same nondominated front, the CD values are set as infinity for extremal solutions and the sum of side lengths of the segment lines that touch the neighbours for the non-extremal solutions. Generally, a solution with a greater CD, which indicates a solution of greater uniqueness, is far from its neighbours in the objective space (Deb et al., 2002b).

3.2.3.2.Hypervolume Contribution

The hypervolume contribution (HVC) selection strategy implements the hypervolume measure proposed by Zitzler & Tiele (1998). The solutions with a greater value of HVC, are those that are normally located within the knee region of a Pareto front (Emmerick et al., 2005). The HVC selection strategy uses nondominated ranks as l, but HVC as v. The HVC for a target solution within a nondominated front ($\boldsymbol{\mu}_l$) is the difference between the hypervolume of all solutions in the front and, the hypervolume of the all solutions in the front that exclude the target solution. In addition, the HVC of the extremal solutions is usually dependent on the reference point selection (Igel & Hansen, 2007). However, in this chapter, the HVC for the extremal solutions are equal to the HVC of the solutions next to each extremal solution, which is the same as Asadzadeh et al., (2014).

3.2.3.3.Hybrid Replacement

The hybrid replacement (HR) selection strategy was first introduced in GALAXY, which is a new hybrid MOEA that tailored for WDS problems (Wang et al., 2017). The HR selection strategy contains two secondary selection criteria of the CD and the ε-dominance criterion (Deb et al., 2005) as v. In the parent selection, the CD is activated for selecting parent solutions for reproducing offspring solutions. However, in the replacement step, the secondary selection criterion being used depends on the number of nondominated solutions in the first front ($\boldsymbol{\mu}_1$). On the one hand, if the number of $\boldsymbol{\mu}_1$ is smaller than *N*, CD will be activated. On the other hand, the ε-dominance selection is activated when the number of $\boldsymbol{\mu}_1$ is greater than *N*. This forces the ε-nondominated solutions to be included in $\boldsymbol{x}_{t+1}$. For the later criterion, the objective domain that is bounded by the two solutions on the tail of $\boldsymbol{\mu}_1$ is divided into (N-2)x(N-2) grids. Each grid is denoted as an ε-box. It is noted that the dominant region within an ε-box, which is at the right bottom corner in this chapter (i.e. maximise resilience and minimise cost objective values). In each ε-box, solutions are found that have the smallest Euclidean distance to the dominant corner and recognise it as an ε-nondominated solution. Therefore, the ε-nondominated solutions are found and included in $\boldsymbol{x}_{t+1}$. The remaining slots left in $\boldsymbol{x}_{t+1}$ are filled by the remaining ε-dominated solutions that are ordered by the distance to the corresponding dominant corner.

3.2.3.4.Random

RND selection strategy was used as a control selection strategy in this chapter. It uses the nondominated fronts as l. The v for each solution are randomly sampled in the range [0, 1], removing any bias in the solution ordering.

3.2.4. Case Studies

3.2.4.1.Two Objective WDS Optimisation Problem

The WDS optimisation problem involves the selection of pipe diameters for a WDS network to optimise set criteria. This selection aims to achieve the design of the lowest cost network that satisfies performance constrains such as the minimum pressure for each node in the network and constraints on the fluid velocity in each pipe. This type of problem is complex to solve as it is NP-hard, nonconvex, high dimensional, multimodal and nonlinearly constrained (Zecchin et al., 2012).

The objective functions used in this chapter are consistent with the popular ones in water resource (Wang et al., 2015, 2017, Jahanpour et al., 2018), namely the maximising of network resilience and minimising of network cost. The cost objective is given by Eq. (3-2).

$$F_c = a \sum_{i=1}^{n} D_i^b L_i \tag{3-2}$$

where F_c = total network cost, which is determined by pipe diameter D_i and pipe length L_i; a and b = specified cost function coefficient and exponent; $n =$ the total number of pipes in the network. The network resilience objective is given by Eq. (3-3).

$$I_n = \frac{\sum_{j=1}^{m} U_j Q_j (H_j - H_j^*)}{\sum_{r=1}^{N_R} Q_r H_r - \sum_{j=1}^{m} Q_j (H_j^* + z_j)} \tag{3-3}$$

where I_n = the network resilience; $m =$ the total number of demand nodes; Q_j, H_j and H_j^* are, the demand, actual head, and minimum head required at each node j, respectively; N_R = the total number of reservoirs; Q_r, H_r are the actual discharge and actual head at reservoir r; and U_j is an indicator of diameter uniformity for pipes that are connected to node j and is defined by Eq. (3-4).

$$U_j = \frac{\sum_{i=1}^{N_{p,j}} D_{ij}}{N_{p,j} \max\{D_{ij} : i = 1, \ldots, N_{p,j}\}} \tag{3-4}$$

where D_{ij} = the diameter of the pipe i connected to node j; $N_{p,j}$ = the total number of pipes that are connected to node j. Note that a larger value of U_j represents a higher reliability of the network node, since the diameter variations between these pipes are lower overall (U_j =1 when all pipe diameters are identical) (Prasad & Park, 2004).

In this chapter, a set of integer options, ranging from 1 to the number of commercially available sizes, is used as the decision variable. The constraints of the WDS optimisation problem in this chapter are the flow velocity in each pipe and pressure head at each node as specified by each case study. EPANET 2.0 (Rossman, 2000) hydraulic simulation software is used to evaluate the flowrates and pressure heads for each pipe and node respectively to compute constraints and the resilience of network.

3.2.4.2.WDS Case Studies and parameter setup

In order to study the impact of different selection strategies on algorithm performance, six WDS design problems commonly adopted in a wide range of MOEA studies (Wang et al., 2015, 2017, Jahanpour et al., 2018, Zheng et al., 2016) were considered in this work. These were chosen as to provide a range of problem characteristics and sizes (Wang et al., 2015). Table 3-1 provides details of the six WDS case studies, where it can be seen that the number of pipes varies from 21 to 454. The configurations of population size and computational budget (i.e. N_{NFE}) were set consistent with the configuration in Wang et al., (2017), as these were found to provide satisfactory outcomes for the case studies. The population size was set as 100 for the NYT, HAN, FOS, and PES problems, and for the large-scale WDS case studies, MOD and BIN, a population size of 200 was used.

Table 3-1. WDS Case Studies and Population Sizes of the MOEAs Considered in the Paper

Scale	Case study	Number of pipes for each case study	Number of options for each pipe	N_{NFE}	N
Small	New York tunnel (NYP)	21	16	$5x10^4$	100
	Hanoi (HAN)	34	6	$5 x10^4$	100
Intermediate	Fossolo (FOS)	58	22	$1 x10^5$	100
	Pescara (PES)	99	13	$1 x10^5$	100
Large	Modena (MOD)	317	13	$4 x10^5$	200
	Balerma (BN)	454	10	$4 x10^5$	200

3.2.5. Result Assessment

3.2.5.1. End-of-run performance metrics

In order to evaluate MOEA performance, three end-of-run performance metrics, hypervolume (I_{HV}) (Zitzler & Thiele, 1999), generational distance (I_{GD}) (Veldhuizen, 1999) and the ε-indicator ($I_{\varepsilon+}$) (Zitzler et al., 2003), were used to assess the relative performance of the algorithms in this chapter. These metrics effectively capture both the convergence and diversity of an algorithms approximate Pareto-optimal set (approximate Pareto front). I_{HV} is the ratio of the dominated volume of an approximate Pareto front compared with a reference Pareto front representing both the convergence and diversity of solutions. I_{GD} is the average distance between an approximate Pareto front and the reference Pareto front, in terms of evaluating the convergence. $I_{\varepsilon+}$ evaluates the minimum distance required to shift the approximate Pareto front to dominate the reference Pareto front, which measures the convergence and consistency of a solution set.

In order to yield a robust comparison among the algorithms' performance metrics, the one-way Kruskal-Wallis test (Kruskal & Wallis, 1952), with Dunn's D post-test (Dunn, 1964) were implemented to evaluate if a pair of algorithms' end of run data significantly differ from each other. This nonparametric analysis provides a statistical test of whether two or more group data means are equal (Hadka & Reed, 2012; Ameca-Alducin et al., 2018). If the difference is not statistically significant, the pairs' data are assigned as being equivalent. Otherwise, the algorithm with the better metric median value is assigned as the better performing algorithm. The pairwise statistical tests were conducted for all algorithm group pairs, where the number of times an algorithm was recorded as being the better performer was recorded and aggregated to yield the overall rank.

3.2.5.2. Selection Metric

In order to provide insight about how a selection strategy affects MOEA search, a novel visual metric is proposed termed the selection metric. The metric is able to visualize the “hot-spots” of a population set, in terms of the solutions’ selection probability to be selected as a parent solution. The idea of the selection metric is to combine the two selection criteria to a single metric for each solution in a population. The selection metric (*Sel*) is defined for solution x from population $\boldsymbol{x}$ as Eq. (3-5):

$$Sel(x|\boldsymbol{x}) = l_{\text{norm}}(x|\boldsymbol{x}) + v\left(x|\mu_{l(x|\boldsymbol{x})}\right) \quad (3\text{-}5)$$

where $l_{\text{norm}}(x|\boldsymbol{x})$ is the normalized primary selection criterion value defined as Eq. (3-6):

$$l_{\text{norm}}(x|\boldsymbol{x}) = 1 - \frac{l(x|\boldsymbol{x})-1}{l_{max}-1} \quad (3\text{-}6)$$

where l is the primary selection criterion value; l_{max} is the greatest primary selection criterion value; and v is the secondary selection criterion value for population $\boldsymbol{x}$. The reason for this metric is to rank solutions by taking account of the influence from both primary and secondary selection criteria values. In this chapter, the primary selection criterion values are greater than one, and have to be normalized to the range [0,1], which is the same as the secondary selection criterion value range.

3.3. Results and Discussion

3.3.1. The Impact of Different Selection Strategies on MOEA Performance

3.3.1.1. Performance comparison

The rank of the three end-of-run metrics and the average ranks of the general MOEA framework algorithms with the different selection strategies for the six WDS problems are shown in Figure 3-6 as categorical surface plots, where shades from white to black indicate the best to the worst ranks, respectively. In Figure 3-6 (a) to (c), the case studies are presented in ascending order of complexity on the horizontal axis and the general MOEA framework algorithms that are embedded with the different selection strategies are listed on the vertical axis. In Figure 3-6 (d), the first three columns are the average ranks across the six case studies for each algorithm for the associated metrics. In addition, the fourth column is the overall average rank, which assess the performance of each algorithm, across the three metrics' average ranks.

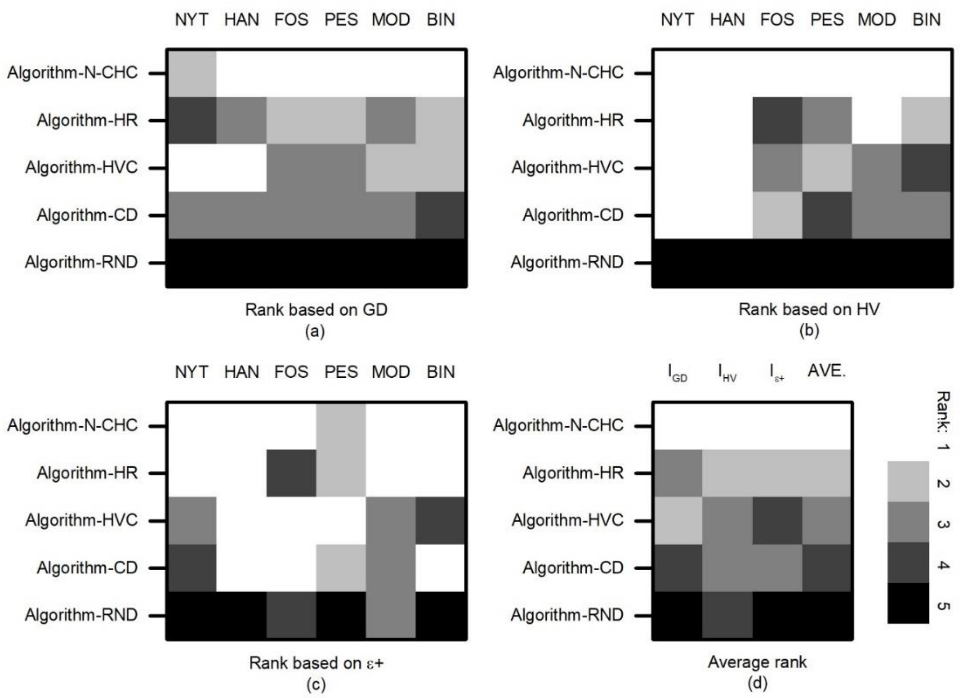


Figure 3-6. Ranks of the general MOEA frameworks with the different selection strategies

As shown in Figure 3-6, it can be consistently observed that the Algorithm-CHC_{Gen} typically performs better than other algorithms with the existing selection strategies, regarding the three end-of-runs metrics as well as the average ranks in Figure 3-6. In contrast, Algorithm-RND remains the worst ranked across the three end-of-run metrics for the six case studies [Figure 3-6. (a)-(c)]. Moreover, Algorithm-HR achieves the second-best average rank as shown in Figure 3-6 (d). This implies the effectiveness of the recently developed HR selection strategy may have been an important contributing factor to the performance of GALAXY (Wang et al., 2017). In addition, Algorithm-HVC outperforms Algorithm-CD, which is consistent with the findings by Igel & Hansen, (2007) [Figure 3-6 (d)]. Given that the only component that is varied in the general MOEA framework algorithms is the selection strategy, these results show that applying different selection strategies affect MOEAs performance, and furthermore, that the CHC_{Gen}

selection strategy outperforms the selected existing selection strategies for WDS problems.

In addition, the benefit of adopting CHC_{Gen} is emphasised as the scale of the problems increase. For example, From Figure 3-6 (b) and (c), apart from Algorithm-RND, the algorithms using the CHC_{Gen}, HVC, CD and HR selection strategies typically achieve similar rank values in small scale problems: NYT and HAN. However, for the large scales problems, such as MOD and BIN, Algorithm-CHC_{Gen} outperforms other algorithms most of the time. This finding suggests that adopting the CHC_{Gen} selection strategy can benefit a MOEA's performance on problems with a high degree of complexity for this type of problem.

3.3.1.2. Discussion

In order to understand how different selection strategies affect an algorithm's search performance, the general MOEA framework with the four selection strategies (i.e. CHC_{Gen}, HVC, HR and CD) was assessed by the selection metric. Figure 3-7 shows the survived population of solutions by applying the four selection strategies to the same combined set $\boldsymbol{c}$, resulted by Algorithm-RND after 20,000 NFE for the BIN problem (selected for illustrative proposes). The categorical colour spectrum indicates the assigned S of the solutions for assessment: from the shading from blue to orange indicates the best to the worst selection metric values, respectively. To help the readers to identify the solutions with higher values, the top 20 ranked solutions are marked by red circles, and are denoted as preferred solutions as they have a higher probability to be selected as parent solutions for reproducing offspring solutions. In this example, overall, the difference of the population of solutions resulted by conducting replacement by the four selection strategies are similar. However, the distribution of the selection metric for each solution are different and discussed. Thereafter, based on the knowledge of the search behaviours resulted from different selection strategies obtained from Figure 3-7, it is possible to gain a deep understanding of the approximate Pareto fronts for the BIN problem that resulted from the general MOEA framework algorithm with different selection strategies, as shown in Figure 3-8.

From Figure 3-7 (a), it can be seen that the CHC_{Gen} selection strategy typically prefers the solutions on the convex hull (typically the solutions with a blue colour), which are located in the low cost and knee region (M€5 to M€8); medium cost range (about M€10.5); and high cost region (about M€20). As shown by the Algorithm-CHC_{Gen} (blue front in Figure 3-8), the approximate Pareto front shows the greatest degree of diversity in resilience value than the general MOEA framework algorithm with other selection strategies (from above 0.45 to over 0.95) and is closer to the reference Pareto front (grey) within the low and mid cost region than other algorithms. However, the lowest ranking selection metric solutions are in the cost range between M€14 and M€18 [Figure 3-7 (a)]. This resulted the inconsistency of the identified solutions along the front within the medium to high cost range [M€14- M€18 in Figure 3-8]. Overall, based on the success of the Algorithm-CHC_{Gen}, it is implied that the portion of the solutions within the convex hull in population set are able to direct the solutions toward the best-known Pareto front, which is consistent with the findings from Jahanpour et al., (2018).

For the HVC selection strategy, the distribution of the preferred solutions [Figure 3-7 (b)] is similar to those resulted by the CHC_{Gen} selection strategy [Figure 3-7 (1)]. However, the HVC selection strategy preferred a more diverse region in the objective space than the CHC_{Gen} selection strategy. For example, the preferred solutions are observed in the cost range between M€8 and M€10; M€14 and around M€18 approximately [Figure 3-7 (b)]. This behaviour results in Algorithm-HVC performing better within the knee region of the Pareto front, as shown in Figure 3-8, than Algorithm-HR, Algorithm-CD and Algorithm-RND. Moreover, the low-ranking solutions of the selection metric are located evenly along the whole front [Figure 3-7 (b)], which prevents the search from missing some areas along the front. The HVC selection strategy resulted in a consistent front (Figure 3-8). As a trade-off, without preferring a particular region in the objective space, Algorithm-HVC fails to achieve the best convergence and relatively poor diversity of the front, in terms of finding solutions only in the narrow resilience value range from above 0.6 to over 0.95.

The difference of the ranking of the solutions by the selection metric values resulted by CD selection strategy and HR selection strategy are subtle, as shown in Figure 3-7 (c) and (d). However, considering closely the population of solutions [the inserts in Figure 3-7 (c) and (d)], it is observed that within each box is an ε-box and the three solutions [excluding the orange solutions in Figure 3-7 (d)] are ε-nondominated solutions [Figure 3-7 (d)]. The three orange solutions with the least CD values are excluded by the CD selection strategy, but preserved within the HR selection strategy as they are the ε-dominated solutions (Deb et al., 2005) and closest to the right bottom corner of the corresponding ε-boxes -the two blocks highlighted by the red colour in Figure 3-7 (d)]. Therefore, with this mechanism, the HR selection strategy is able to preserve good convergence in population. Such attributes as these would enable the population to converge to the global optima effectively (Wang et al., 2017). As shown in Figure 3-8, Algorithm-HR outperforms Algorithm-CD in terms of convergence of its approximate front.

In summary, it is found that augments the selection preference to the solutions on the convex hull of the population set would improve both convergence and diversity of the approximate front. This is somewhat surprising, as these preferred solutions are on the distinct clusters of the approximate front, rather than being on relatively uniformly along the approximate front. The finding implies that focusing on exploiting the solutions on convex hull would improve algorithm performance.

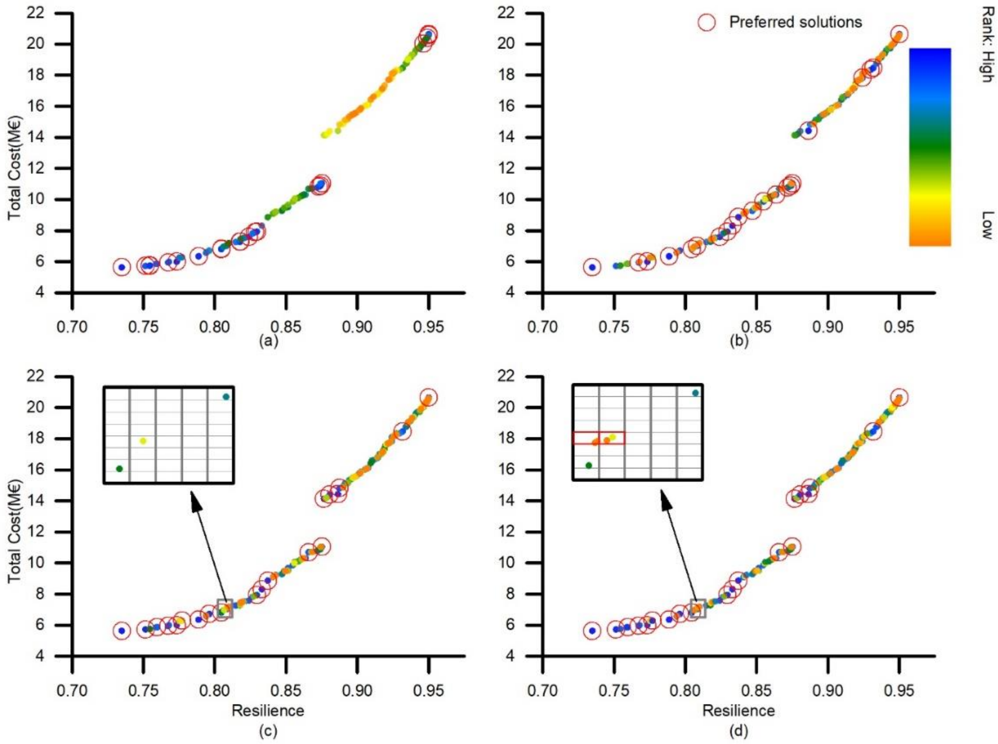


Figure 3-7. Selection metric (for a range of selection strategies) for the population of solutions from an Algorithm-RN run applied to the BIN problem (taken at iteration NFE=20,000). Subfigures are for the selection metric based on the following selection strategies: (a) CHCGen; (b) HVC; (c) CD; and (d) HR. The inserts are zoomed in views of points in the grey boxes in (c) and (d).

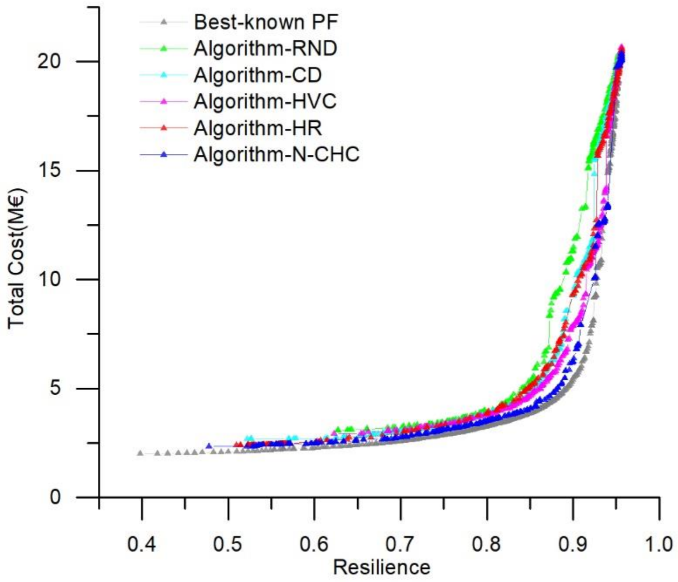


Figure 3-8. Approximate Pareto front solutions for the general MOEA frameworks with different selection strategies.

3.3.2. The Application of CHCGen Selection Strategy to Existing MOEAs

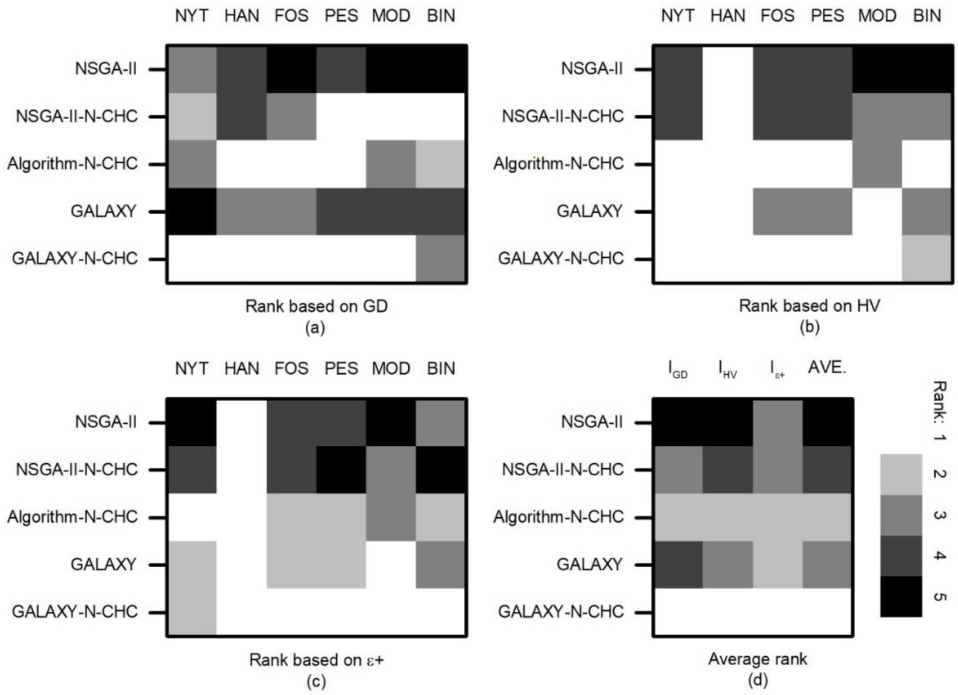


Figure 3-9. Ranks of NSGA-II, NSGA-II-NCHC, GALAXY, GALAXY-CHC$_{Gen}$ and the general MOEA framework with CHC$_{Gen}$ selection strategy.

Figure 3-9 shows the algorithm ranks according to the end-of-runs metrics for two existing generational MOEAs (NSGA-II and GALAXY), and these two MOEAs with the adoption of the CHC$_{Gen}$ selection strategy (along with Algorithm- CHC$_{Gen}$). As can be seen, the CHC$_{Gen}$ benefits existing MOEAs. This is indicated by the fact that NSGA-II-CHC$_{Gen}$ and GALAXY-CHC$_{Gen}$ outperform NSGA-II and GALAXY [Figure 3-9 (d)], respectively. Moreover, the results show that the influence of the selection strategy does not outweigh other search strategies. This is illustrated by comparing the relative influence of the hyperheuristic and the operator set with the selection strategy, respectively. On the one hand, the influence of the hyperheuristic dominates the selection strategy. This is demonstrated by GALAXY outperforming

Algorithm-CHC_{Gen} [Figure 3-9 (d)]. On the other hand, the influence of the number of operators outweighs the selection strategy. This is evidenced by the higher average rank of Algorithm-CHC_{Gen} in comparison to NSGA-II-CHC_{Gen} [Figure 3-9 (d)]. In addition, the results show the benefit of implementing an adaptive hyperheuristic and using larger operator set size. This is demonstrated by GALAXY-CHC_{Gen} outperforming Algorithm-CHC_{Gen}; and Algorithm-CHC_{Gen} outperforming NSGA-II-CHC_{Gen}, respectively [Figure 3-9 (d)].

3.4. Conclusion

A novel selection strategy called the generational convex hull contribution (CHC_{Gen}) is developed in this chapter. The CHC_{Gen} selection strategy not only accounts for convergence and diversity in generating the approximate front, but its search behaviour is well suited for problems that have a convex shape of the approximate fronts. Moreover, the selection strategy is desired to result in better solutions being identified within the 'knee regions' of approximate Pareto fronts.

CHC_{Gen} was compared with four existing selection strategies by implementing these strategies within a consistent general MOEA framework. The general MOEA framework with the CHC_{Gen} selection strategy was found to outperform four other popular existing selection strategies in the numerical study involving six WDS problems. The CHC_{Gen} selection strategy showed the overall best convergence, diversity and consistency of the approximate fronts that were generated.

In order to understand that how the influence of the selection strategies reflect on the attributes of the approximate front, each algorithm with different selection strategies were applied to search from a common population set for the most complicate problem – BIN. It was observed that the CHC_{Gen}

selection strategy augments the selection preference to bias the population solutions that lie on the convex hull regions of the approximate front. Given the nature of the convex hull solutions within a nondominated set that are closer to the "ideal point" and in distinct regions along the approximate front, this type of selection preference leads to an improved convergence and diversity of the search. Therefore, CHC_{Gen} selection strategy allows the algorithm effectively explore the search space and results in the best performance of the approximate front in comparison to other existing selection strategies.

To further investigating the potential of the CHC_{Gen} selection strategy to improve existing MOEAs, the current best generational MOEA on solving WDS problems (GALAXY) and the industry standard generational MOEA (NSGA-II) were modified to incorporate the CHC_{Gen} selection strategy. The CHC_{Gen} selection strategy was found to be able to boost the performance of these two algorithms, suggesting that the proposed selection strategy could benefit other existing MOEAs.

Chapter 4 Multi-Objective Evolutionary Algorithm Component Investigation for Solving Water Distribution System Optimisation Problems

Peng Wang[1,2], Aaron C. Zecchin[1], Holger R. Maier[1,2]

[1]School of Civil, Environmental and Mining Engineering, The University of Adelaide, Adelaide, SA, Australia;

[2]College of Civil Engineering and Architecture, Zhejiang University, China

Statement of Authorship

Title of Paper	A Multi-Objective Evolutionary Algorithm Component Investigation for Solving Water Distribution System Optimisation Problems	
Publication Status	☐ Published ☐ Submitted for Publication	☐ Accepted for Publication ☑ Unpublished and Unsubmitted work written in manuscript style
Publication Details		

Principal Author

Name of Principal Author (Candidate)	Peng Wang		
Contribution to the Paper	Primary innovator, analyst and author Conception and design of the project Development and execution of numerical experimental program Analysis and interpretation of research data Draft the paper		
Overall percentage (%)	90		
Certification:	This paper reports on original research I conducted during the period of my Higher Degree by Research candidature and is not subject to any obligations or contractual agreements with a third party that would constrain its inclusion in this thesis. I am the primary author of this paper.		
Signature		Date	01/10/2020

Co-Author Contributions

By signing the Statement of Authorship, each author certifies that:

i. the candidate's stated contribution to the publication is accurate (as detailed above);

ii. permission is granted for the candidate in include the publication in the thesis; and

iii. the sum of all co-author contributions is equal to 100% less the candidate's stated contribution.

Name of Co-Author	Aaron C. Zecchin		
Contribution to the Paper	Conception and design of the project Analysis and interpretation of research data Critically revising the paper so as to contribute to the interpretation		
Signature		Date	17/07/2020

Name of Co-Author	Holger R. Maier
Contribution to the Paper	Conception and design of the project Analysis and interpretation of research data Critically revising the paper so as to contribute to the interpretation

Signature		Date	

Abstract

Multi-objective evolutionary algorithms (MOEAs) have been regarded as effective optimisation tools for solving water distribution system (WDS) problems for over 20 years. The components of MOEAs are the key factors affecting the algorithms' search behaviours, thereby affecting algorithm performance. Traditionally, to propose an effective MOEA, many works diligently propose new algorithm components. Currently, fine-tuning the components is a sufficient and effective method to improve an MOEA's performance. In this chapter, a systematic investigation is conducted to identify the optimal component combination and propose a general multi-objective evolutionary algorithm. Many popular and state-of-the-art components were considered in this chapter. Moreover, the impact of different components on algorithm performance was also studied comprehensively. In addition, the effectiveness of the general multi-objective evolutionary algorithm is assessed by comparing seven existing MOEAs by solving six WDS problems.

Keywords: Multi-objective evolutionary algorithms, water distribution system design optimization

4.1. Introduction

Water distribution system (WDS) optimisation problems are framed around the design of a pipe network at minimal cost, which satisfies the network's hydraulic constraints (Jetmarova et al. 2018; Keedwell and Khu 2003; Prasad and Park 2004). The characteristics of this problem type are that it is non-linear and multimodal, which results in difficulty in finding efficient, global, optimal solutions (Zecchin et al. 2012; Zheng et al. 2017). In the last 30 years, evolutionary algorithms (EAs), particularly multi-objective evolutionary algorithms (MOEAs), have been proposed and have experienced growing popularity in applications for WDS problems (Jahanpour et al., 2018; Wang et al., 2015; 2017; Zecchin et al., 2012; Zheng et al., 2016). Traditionally, the development of an MOEA is focused on key algorithmic improvements, addressing limitations in computational efficiency and algorithm search quality. For example, Deb et al., (2002b) proposed NSGA-II, which aimed to: (i) reduce the high computational complexity of nondominated sorting; (ii) achieve elitism preservation; and (iii) develop an effective diversity maintenance strategy. These objectives were fulfilled by the fast-nondominated sorting approach, comparing offspring with the population to retain elite solutions and use of the crowding distance metric, respectively.

As algorithm performance is a combination of not only the internal workings of an algorithm, but also the problem characteristics, Vrugt et al., (2007) found that the nature of a fitness landscape often varies considerably in different problems and proposed AMALGAM in response. AMALGAM is a self-adaptive algorithm, designed to respond dynamically to strategies that are found to be most effective in the given search space. Inspired by models of adaptation in natural systems, AMALGAM was developed to use multiple search operators assisted by a hyperheuristic to tune each operator's utilization during the search (Vrugt et al., 2007). The key aspect in this work is the hyperheuristic, which allows effective feedback concerning which operators are likely to be most effective for the search. Hadka and Reed (2013) studied the weaknesses of existing MOEAs comprehensively for high dimension optimisation problems, which include: (i) an appropriate nondominance relationship for high dimension objective numbers; (ii) an appropriate diversity maintenance strategy, (iii) a reduction in the risk of deterioration; and (iv) a reduction in the works for parameterization. Borg was developed to include many new search strategies such as ε-nondominance; multiple operators with a hyperheuristic; ε-progress and an adaptive population sizing operator (hyperheuristic) to overcome the difficulties mentioned above. The algorithms listed above have shown effective performance not only in a wide range of applications and test functions (Asadzadeh and Tolson 2012; Hadka and Reed, 2012, 2013, 2015; Zeff et al. 2016; Zhang et al. 2010), but also in WDS problems (Wang et al., 2015; 2017; Zheng et al., 2016). Following on from these works, Wang et al., (2017) aimed to improve MOEA performance by customising existing search strategies. This is achieved by tailoring operators for WDS problems (i.e. the discrete search space); developing a new hybrid replacement selection strategy to maintain solutions' convergence and diversity; and applying a hyperheuristic that is akin to AMALGAM to adapt the search to various problems' characteristics. It has been demonstrated that GALAXY outperformed the aforementioned algorithms on a range of WDS problems (Wang et al., 2017)

In addition to the development of different algorithmic formulations, in order to improve algorithm performance, many studies have been conducted to refine an MOEA's components for a given algorithm. For example, Vrugt et al., (2009) studied the impact of operator combinations on AMALGAM's performance to find the best operator combination from a set of 5 operators. Wang et al., (2020) investigated the impact of the number of operators on algorithm performance by comparing an algorithm with a different number of operators and suggested a large operator size to be of benefit for algorithm performance. Moreover, Matthew et al., (2013) found Borg is relatively insensitive to its parameter settings by conducting massive numerical experiments, indicating a robustness of performance for this algorithm. In addition, (Wang et al., 2020b) studied the impact of different selection strategies by comparing a general MOEA framework separately incorporating five selection strategies. These works provide a more insightful understanding as to which algorithm's search strategies are more effective and provide insight for designing MOEAs in the future.

From previous studies, it is clear that different search components embedded in an MOEA affect algorithm performance, which is evident for the case of two algorithms sharing the same structure of processes (i.e. parental selection, reproduction and off-spring selection) but with different components within those processes (e.g. AMALGAM and GALAXY use different operator sets for reproduction). The reason is that the different components' characteristics are unique, thereby resulting in a different search behaviour. Therefore, this leads to the observation that optimizing the components within an algorithm framework presents an opportunity to fine tune an MOEA's search behaviour and improve its performance. However, past work has typically only focused on either entirely different algorithm structures, or singular processes in isolation (e.g. only parental selection). To the authors' knowledge, a comprehensive analysis involving a systematic investigation into simultaneously varying the components across the range of algorithmic processes is still lacking. Consequently, the objective of this chapter is that within a common MOEA framework, a systematic numerical experiment is conducted to investigate the impact of component combinations on algorithm performance. In the second objective, the best performing component combination MOEA is selected and compared with seven existing MOEAs on WDS problems.

To achieve the above objective within this chapter, firstly, a comprehensive systematic investigation is conducted using the general multi-objective evolutionary algorithm (GMOEA) framework (Wang et al. 2020) to construct algorithms automatically, based on varying the components of the framework's processes (e.g. the operator set and hyperheuristic for reproduction, and the selection strategy for the population selection processes). This investigation not only yields insight into the influence of singular and pair-wise variations of components, but also allows for the systematic construction of the best performing algorithm across the range of algorithms considered. This algorithm is symbolised by the notation GMOEA(CHC_{gen},12,T,A), showing that it adopts the GMOEA framework with a new convex hull contribution selection strategy (the CHC_{gen} in the parenthesis), an operator set of 12 operators (12), the transitional hyper-heuristic (T), and the archive meta-data (A) (all of these components are explained in later sections). The algorithm GMOEA(CHC_{gen},12,T,A) demonstrates great performance on the six WDS problems and has identified a significant number of new best-known Pareto Front solutions for the more complex problems.

4.2. Methodology

4.2.1. Overview

An overview of the proposed experimental methodology is outlined in Figure 4-1. In order to investigate the impact of MOEA components on algorithm performance, a GMOEA framework (Wang et al. 2020) was outlined that contains the key processes of selection (i.e. parent selection and survivor selection) and reproduction, into which feed the components of selection strategy, along with the operator and hyperheuristic components, which is shown in the top-left block in Figure 4-1. To achieve a comprehensive investigation into the influence of different components, a range of process component alternatives are analysed through the GMOEA framework as constructed GMOEAs. Through comparing the performance of these constructed algorithms (i.e. algorithms constructed where specific components are selected for each process), it is possible to study how each component, individually and in a pair-wise sense, affect algorithm performance. Adopting this strategy, the best component combination is able to be identified and a final GMOEA is proposed. Thereafter, in order to assess the performance of the proposed GMOEA, it was compared with eight existing MOEAs, with a full computational budget, to evaluate how many solutions from each algorithm contribute to the best-known Pareto front solution set.

The above MOEAs were tested by solving six WDS case studies with different characteristics that are associated with the size of the search spaces (Wang et al., 2015). These are outlined in the central block in Figure 4-1. To maintain a practical computational time, the partial computational budget proposed by Wang et al., (2017) (sufficient to guarantee convergence of algorithms) was used in this part of the study. Then, the best performing MOEA was selected as the proposed MOEA (top-right block in Figure 4-1).

Thereafter, the proposed MOEA was compared with eight state-of-the-art MOEAs with a full computational budget to demonstrate the effectiveness of the proposed MOEA. In addition, within the above numerical experiments, 30 independent runs were undertaken for each algorithm application.

In the result assessment block at the bottom of the Figure 4-1, three end-of-run metrics were used to assess the performance of the 20 MOEAs with a non-parametric statistical test to report any statistical difference of the metrics among the MOEAs. For objective 2, as many new best-known Pareto front solutions have been found by the proposed MOEA, the metric "percentage of contribution" to best-known Pareto fronts was used to compare the proposed MOEA with the other state-of-the-art MOEAs.

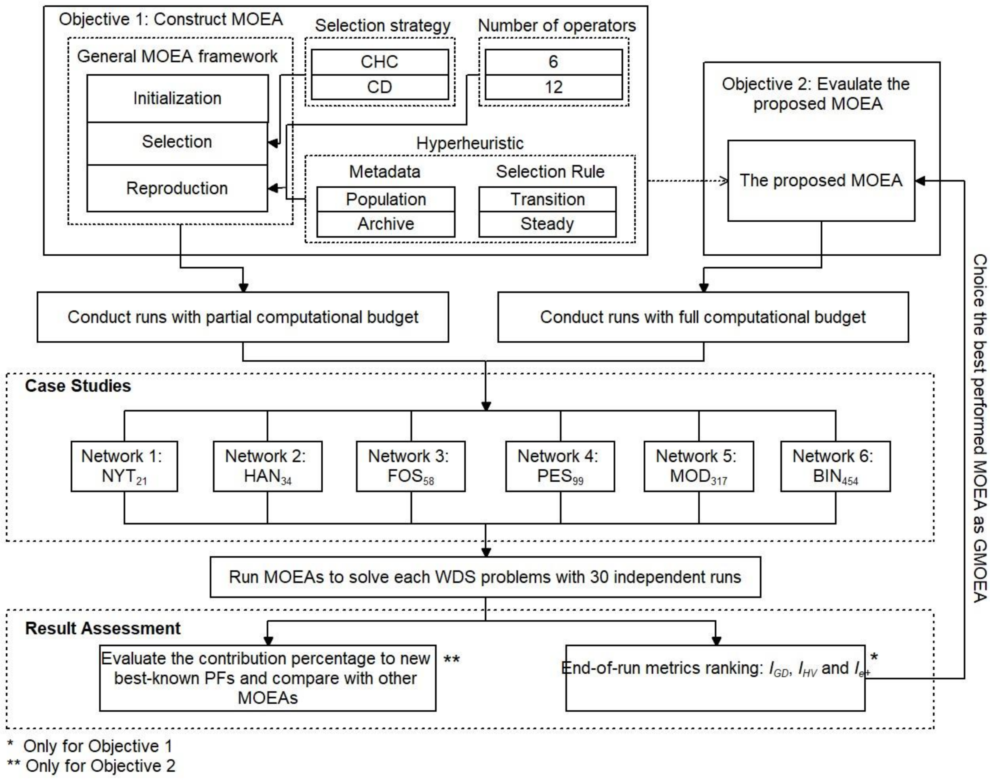


Figure 4-1. Flowchart of the methodology for each objective

4.2.2. General MOEA Framework

In order to study the impact of the MOEA components on algorithm performance systematically, a process of MOEA, shown in Figure 4-2, was constructed by summarizing a range of state-of-the-art MOEAs' features. Starting from initialisation, a set of solutions (or population) is produced randomly. Some population members are selected as parent solutions, which are identified by a selection process that then feeds into the reproduction process. After reproduction, the survivor selection strategy identifies the successful offspring. These successful offspring replace the unsuccessful population solutions and form the new population set for the next generation. The above processes are repeated until a termination criterion is met [e.g. running out of number of function evaluation (NFE)]. The key components of each of these processes are shown in the dash-line boxes in Figure 4-2. They each induce a unique search behaviour, thereby affecting population, parent and offspring characteristics. The process components are introduced as follows:

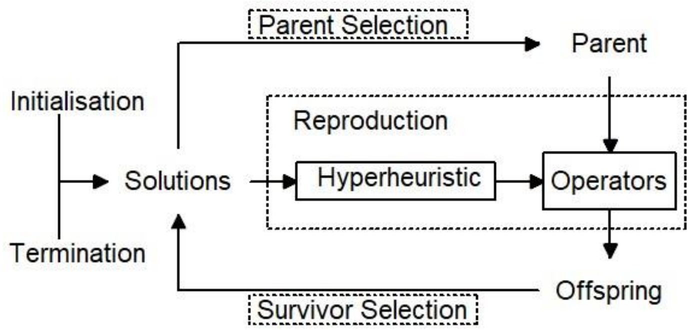


Figure 4-2. Steps for the General MOEA framework

The parent and survivor selection processes are characterised by the selection strategy component. It is a key component of an MOEA and affects the algorithm's performance. This is because the selection strategy is the component that mainly determines the character of the population (Back, 1996; Hanne, 1999). The selection strategy determines which solutions are recognized as "good" and have an opportunity to produce offspring in the parent selection process. The survivor selection improves the population's fitness by selecting the successful offspring to include in the solution set.

The reproduction process involves two components, as shown in Figure 4-2: the hyperheuristic and the operator set. The operator set contains low-level operators, which are search strategies that produce an offspring (i.e. through recombination or randomisation processes), thereby allowing the algorithm to explore the search space. The utilisation of operators in this set is managed by a hyperheuristic, which is a self-adaptive scheme for selecting the most appropriate lower-level operators (Burke et al., 2013; Drake et al., 2019). The hyperheuristic plays an important role in MOEA performance as it controls the degree to which operators contribute to produce offspring solutions in each generation. In the next section, the actual working of these components, and some of examples that have been implemented in state-of-the-art MOEAs, are discussed.

4.2.3. MOEA Components

4.2.3.1. Selection Strategy

As outlined above, the selection strategy (employed in both the parent selection, or survivor selection processes) is a process of determining which solutions are selected to be included in the parental and next generation population sets. For a generational MOEA, selection is determined by primary and secondary selection criteria, with the secondary criterion only considered when two solutions have the same value of the primary selection criterion. The primary selection criterion is typically based on the non-dominance status of the solutions (i.e. whether the solution is a member of the primary or higher order non-dominance front), and the secondary criteria is based on solution diversity metrics (Wang et al., 2020b). In this chapter, two selection strategies for the secondary criteria were considered. The first strategy, crowding distance (CD), is the most popular strategy that has been applied to various MOEAs such as NSGA-II, AMALGAM, ε-NSGA-II and GALAXY. The second strategy is the convex hull contribution (CHC_{gen}) selection strategy that was recently developed for generational MOEAs and found to outperform other existing selection strategies. The details of the two selection strategies are outlined as follows:

Crowding distance selection strategy

The crowding distance (CD) selection strategy is a popular selection strategy that has been widely applied to many MOEAs owing to its simplicity and effectiveness since its conception by Deb et al., (2002b). The primary selection criterion is the non-dominance status of the solutions, where solutions x_i with a higher non-dominance rank outweigh others with a lower non-dominance rank. CD is the secondary selection criterion, and is used to find and discard solutions with the smallest CD within the same non-dominance rank, set to maintain population size. Within a non-dominated set ($\boldsymbol{ND}_l$, which is the set of solutions in the l-th non-dominance front) in the population, the solutions are sorted based on any one objective value. The metric is equal to infinity for the extremal solutions (i.e. for $i=1$ *or* $i=|\boldsymbol{ND}_l|$), which have the greatest and smallest objective values f_{max}^m and f_{min}^m across all m objectives, respectively. For a non-extremal solution $x \in \boldsymbol{ND}_l$, the CD value of the solution is given by the sum of the side lengths of the segment lines that touch the neighbouring solutions to x (Deb et al., 2002b) and can be given by:

$$CD(x|\boldsymbol{ND}_l) = \sum_{m=1}^{M} \frac{f_{x_{i+1}}^m - f_{x_{i-1}}^m}{f_{max}^m - f_{min}^m} \tag{4-1}$$

where f_x^m is the m-th objective value for solution x, and $\langle x,m \rangle$ is used to denote the nearest neighbour solutions to x in $\boldsymbol{ND}_l$ with respect to objective m.

New convex hull contribution selection strategy

The new convex hull contribution (CHC) selection strategy for generational MOEAs was proposed by Wang et al., (2020b) to improve algorithm performance in water resource problems that have convex hull shaped Pareto fronts. A set is called convex if, for every two solutions inside the set, all solutions on the line segment between them are inside the set (Asadzadeh, et al. 2014). A convex hull is a set of points that are the intersection of all convex sets containing those points (Barber et al. 1996). For a solution x in a convex hull set, its convex contribution is the difference of the convex hull size (i.e. length, area, or hypervolume in one, two, or higher dimensional spaces, respectively) between the set with and without x. For example, in the bi-objective domain, the step involved in a CHC evaluation is described as follows: firstly, the area covered by the entire convex hull set of solutions is evaluated. Then, the area of the set that excludes x is evaluated. In the end, the CHC value of x is the area difference of the convex hull set with and without x.

Given the fact that many CHC values in a MOEA's population set are zero (Asadzadeh, et al. 2014), the CHC_{gen} uses the fast non-dominance sorting approach and convex hull contribution (CHC) to assign a Convex Hull rank $\boldsymbol{CH}$ and a non-zero CHC to each population to preserve elitism within the population. The procedure of the CHC_{gen} selection strategy is outlined as follows: within the first non-dominated set in the population ($\boldsymbol{ND}_1$), the CHC values are evaluated for the set's solutions, and the solutions in $\boldsymbol{ND}_1$ with non-zero CHC values are assigned a CH rank of one. Then, the remainder of the solutions in $\boldsymbol{ND}_1$ are evaluated (based on the $\boldsymbol{CH}_1$ solutions being omitted from this evaluation), and the solutions with a non-zero CHC value are assigned as $\boldsymbol{CH}_2$. The above procedure is iterated until all the solutions in $\boldsymbol{ND}_1$ have been assigned a $\boldsymbol{CH}$ and possess a non-zero CHC value. Thereafter, the same procedure is carried out for the remainder of the non-dominated sets.

The CH ranks are hierarchically ordered with respect to each non-dominance front. That is, the set of solutions in the CH rank two set of $\boldsymbol{ND}_1$ rank more highly in terms of selection than the CH rank one set of solutions from $\boldsymbol{ND}_2$. Hence, the primary selection criterion represented by the CH rank sets preserve the elitism of the non-dominance. The secondary selection criterion, being the CHC value within each CH rank set, indicates solution diversity. CHC_{gen} will favour solutions with a higher CH rank and greater CHC values, which represent better convergence and diversity.

4.2.3.2.Operator Set

For an MOEA with multiple operators, performance is a matter of not only the operator combination, but also the number of operators. The operators used in this chapter are outlined in Table 4-1. To select appropriate operators, two schemes are considered. The first one is the operator combination that is used in GALAXY, as these operators were specially designed for WDS problems and have shown effective performance against other state-of-the-art MOEAs (Wang et al., 2017). The second scheme includes the 12 operators outlined in Table 4-1. These have been proven to outperform the other operator combinations of the state-of-the-art MOEAs (Wang et al., 2020a).

Table 4-1. Operators Applied in the Computational Experiments

Operators	
Simulated binary crossover[a] (SBX)	Uniform mutation for integer[f] (UMI)
Differential evolution[b] (DE)	Gaussian mutation for integers[g] (GMI)
Parent-centric crossover[c] (PCX)	Dither creeping for integers[h] (DCI)
Unimodal normal distribution crossover[d] (UNDX)	Differential evolution for integers[b] (DEI)
Simplex crossover[e] (SPX)	Simulated binary crossover for integers[g] (SBXI)
Polynomial mutation[a] (PM)	Turbulence factor for integers[i] (TFI)

Notes: [a]Deb & Agrawal., (1994); [b]Storn & Price, (1997); [c]Deb et al. (2002a); [d]Kita et al. (1999); [e]Tsutsui et al. (1999); [f]Michalewicz, (1992); [g]Wang et al., (2017); [h]Zheng et al. (2013); [i]Pulido et al. (2004).

4.2.3.3.Hyperheuristic

According to the 'No-free-lunch theorem' (Wolpert and Macready, 1997), it is impossible to develop a single MOEA that is universally the most efficient for all optimization problems, given different algorithms have different search behaviours that are only efficient for some problems. In past research, operators have been shown to play a dominant role in affecting an MOEAs' search behaviour (Wang et al. 2020). Inspired by models of adaptation in natural systems (Vrugt et al. 2007), many MOEAs using multiple operators with a high level hyperheuristic have been proposed and have demonstrated effective performance on a range of problems (Hadka and Reed, 2013; Vrugt et al. 2009; Wang et al. 2017). For an MOEA with multiple operators, a hyperheuristic is a process that automatically selects operator(s) to produce current solutions within each generation of the search (Drake et al., 2019). A hyperheuristic involves a feedback loop in which operators that produce more successful offspring are rewarded by increasing the number of offspring produced by that operator (Hadka and Reed, 2013). Within the feedback loop, there are two components: the selection rule and the metadata. Through a streaming process (i.e. passing the metadata to a selection rule in each generation), hyperheuristics are able to make a decision as to which and how many operators produce offspring in the reproduction process. In this chapter, two popular hyperheuristics, used in Borg and AMALGAM, were considered and the two components for each of them are described as follows:

Metadata

The metadata are properties from a set of solutions that provide feedback to the selection rule to update the probabilities of each operator being able to produce offspring. The metadata $\boldsymbol{Z}_t$ is given as a set of triples of the form ($\boldsymbol{x}$, $\boldsymbol{f}(\boldsymbol{x})$, j) where $\boldsymbol{x}$ is a solution set included in the metadata at iteration t; $\boldsymbol{f}(\boldsymbol{x})$ contains its objective values, and j is the operator from which $\boldsymbol{x}$ was generated. The set of solutions included in the metadata can vary from algorithm to algorithm. For example, for Borg's hyperheuristic, the metadata is the current ε-non-dominance solutions (Archive Metadata); and for AMALGAM's hyperheuristic, the metadata is the solution contributing to the population set (Population Metadata). In this chapter, two types of metadata, 'Archive' and 'Population', were included in the numerical experiments.

Selection rule

Selection rule is associated with the selection of operators prior to reproducing offspring solutions that are likely to, themselves, produce successful offspring solutions that approach the true Pareto front solutions. The basic formula of the selection rule is outlined as:

$$P_j^t = \frac{c_j^t}{\sum_{i=1}^{k} c_i^t} \tag{4-2}$$

where P_j^t is the selection probability of operator j; c_j^t is a non-zero factor that indicates the degree of success of operator j; and k is the number of operators. P_j^t is categorized as either transition-based or steady state-based in different

MOEAs. For example, Borg uses a steady state selection rule where by c_j^t is based on the size of components in the metadata, given by:

$$c_j^t = N_j(\boldsymbol{Z}_t) + \varsigma \quad (4\text{-}3)$$

where $N_j(\boldsymbol{Z}_t)$ represents the number of solutions in $\boldsymbol{Z}_t$ that were generated using operator *j*; and $\varsigma = 1$ is used to prevent any selection probability from reaching zero. Thus, the steady-state selection probability is calculated by the proportion of the solutions in the metadata. The transition state selection rule, adopted in AMALGAM, defines c_j^t as the ratio between the size of the relevant solutions in the metadata and the quota of operator j of the last generation (symbolised as N_j^{t-1}) and is given by:

$$c_j^t = \frac{N_j(\boldsymbol{Z}_t)}{N_j^{t-1}} \quad (4\text{-}4)$$

where $N_j^{t-1} = \max\{1, \lceil N \cdot P_j^{t-1} \rceil\}$ is the total number of solutions generated in iteration *t*-1 using operator *j*. It is noted that $N_j^{t-1} = 1$ only if $P_j^{t-1} = 0$ to avoid the possibility of inactivating operators that may contribute to the search in future generations.

4.2.4. Case studies

4.2.4.1.Objective functions

In order to compare the proposed MOEA with other existing MOEAs that were studied in previous works, this study used the same objective functions that are popular in WDS problems (Wang et al., 2015, 2017, Jahanpour et al., 2018). The first objective function is to minimize the cost F_c):

$$F_c = a \sum_{i=1}^{n} D_i^b L_i \qquad (4\text{-}5)$$

where D_i and L_i = diameter and length for pipe i, respectively; a and b = constants that are associated with different problems; $n =$ the total number of pipes in the network. The second objective is to maximize the network resilience (I_n), as proposed by Prasad and Park, (2004), which measures the combined effects of surplus power and nodal uniformity.

$$I_n = \frac{\sum_{j=1}^{nd} \left(\frac{\sum_{i=1}^{N_{p,j}} D_{ij}}{N_{p,j} \max\{D_{ij}: i=1,\ldots,N_{p,j}\}} \right) Q_j (H_j - H_j^*)}{\sum_{r=1}^{N_R} Q_r H_r - \sum_{j=1}^{m} Q_j (H_j^* + z_j)} \qquad (4\text{-}6)$$

where $nd =$ the total number of demand nodes; $N_{p,j} =$ the total number of pipes that are connected to node *j*; D_{ij} = the diameter of pipe *i* connected to node *j*; Q_j, H_j and H_j^* = the demand, actual head, and minimum head required at each node *j*, respectively; $N_R =$ the total number of reservoirs and Q_r, H_r = actual discharge and actual head at reservoir r, respectively.

In each WDS case study there exist hydraulic constraints that must be met. For example, a node's head should be greater than the corresponding minimum head required. Also, for some cases (e.g. Fossolo, Pescara and Modena problems), the constraints involve minimum and/or maximum flow velocities. The interested reader can refer to Wang et al., (2015) for further details.

The objective functions were evaluated by running the hydraulic software EPANET 2 toolkit in the software language C, complied to a MEX function that is available for MATLAB source code. In addition, in order to handle the infeasible solutions, the degree of constraint violation is recorded for each solution and the constraint-tournament selection (Deb et al., 2002b) was applied to the all algorithms tested in this chapter.

4.2.4.2.WDS Study Networks

In this paper, six WDS case studies that are widely used to compare the performance of different MOEAs in other works (Jahanpour et al., 2018, Wang et al., 2015, 2017, Zheng et al., 2016) were considered and are shown in Table 4-2. They are categorized into three different types according to the number of pipes, varying from 21 to 454. The networks New York tunnel (NYT) and Hanoi (HAN) have 21 and 34 pipes, respectively, and are classified as small scale problems; Fossolo (FOS) and Pescara (PES) have 58 and 99 pipes, respectively, and are classified as medium scale problems; Modena (MOD) and Balerma (BIN) have 317 and 454 pipes, respectively, and are classified as large scale problems. The numbers of the partial NFEs for each case study are consistent with the setting in Wang et al. (2017) and Wang et al. (2020), which are sufficient for the MOEAs to converge. In addition, to achieve a more extensive comparison with the existing MOEAs outlined in Wang et al. (2015) and Jahanpour et al. (2018), the full NFE and epsilon precisions for cost (EF_c) and network resilience (EI_n), as adopted in Wang et al., (2015), were used.

Table 4-2. WDS Problems Specifications

Scale	**Case study (problem)**	**Number of options for each pipe**	***NFE* (Partial)**	***NFE* (Full)**	EF_c	EI_n	***N***
Small	New York tunnel (NYP_{21})	16	$5x10^4$	$6x10^5$	10^0	10^{-3}	100
	Hanoi (HAN_{34})	6			10^{-1}	10^{-3}	
Medium	Fossolo (FOS_{58})	22	$1x10^5$	$1x10^6$	10^{-3}	10^{-3}	100
	Pescara (PES_{99})	13			10^{-2}	10^{-3}	
Large	Modena (MOD_{317})	13	$4x10^5$	$2x10^6$	10^{-2}	10^{-3}	200
	Balerma (BN_{454})	10			10^{-2}	10^{-3}	

4.2.5. Numerical experiment setup

In order to study how each component affects algorithm performance, as outlined in Section 4.2.1, a comprehensive investigation that focuses on comparing different component combinations of the constructed GMOEA was conducted. Table 4-3 outlines the alternatives for each component that is considered in this chapter (also shown in Figure 4-1). A total of 20 component combinations (i.e. constructed realisations of the algorithm) were compared by optimising the six WDS problems that are outlined in Section 2.3.2. The name of each constructed MOEA is given as by following example: GMOEA(CHCgen,12,T,A) is the algorithm with a CHC_{gen} selection strategy, 12 operators, transitional probability selection rule, and the archive metadata. The results of the algorithms with the 20 component combinations were evaluated by three end-of-runs metrics and post-processed by a non-parametric statistical analysis that is outlined in section 4.2.6.1. To study the impact of each component on algorithm performance, the performance of the algorithms with the same component within a given process were grouped and compared with the performance of other grouped constructed GMOEAs with alternative components within the given process. Moreover, to investigate the influence of each pairwise component combination, the results of the algorithms with the same two components were grouped together and compared against the other alternatives. Conducting this investigation helps to understand not only how each component alternative affects algorithm performance, but also how pairwise combinations affect algorithm performance. The performance of the algorithm with each component combination was evaluated and the best performing algorithm was selected.

Table 4-3. Search Components Varied in the Computational Experiments

Selection Strategy	Number of Operators	Hyperheuristic	
		Metadata	Selection Rule
CHC_{gen}	6	Archive (A)	Transitional Probability (T)
CD	12	Population (P)	Steady Probability (S)

In order to propose a new MOEA by finding the combination process component that yields the best performing algorithm, GMOEA(CHCgen,12,T,A) is selected from the best performing 20 algorithms with different component combinations. To evaluate the performance of GMOEA(CHCgen,12,T,A) with other state-of-the-art MOEAs, Jahanpour et al., (2018) and Wang et al., (2015) conducted extended optimization to the WDS problems with full computational budgets (as outlined in Table 4-2) for eight state-of-the-art MOEAs, including: NSGA-II, ε-MOEA, ε-NSGA-II, AMALGAM, Borg, GALAXY, PADDS-CHC and PADDS-HVC. The Pareto fronts for these algorithms with full computational budgets are used within this paper and are available from Jahanpour et al., (2018) and Wang et al., (2015). In this chapter, the approximate sets of GMOEA(CHCgen,12,T,A) were collected to update the best-known Pareto fronts of the WDS case studies.

4.2.6. Results assessment

4.2.6.1. End of run metrics

In order to evaluate each MOEA's performance, three end-of-run performance metrics were used to assess the relative performance of the algorithms in this chapter, namely hypervolume (I_{HV}) (Zitzler and Thiele, 1999), generational distance (I_{GD}) (Veldhuizen, 1999) and ε-indicator ($I_{\varepsilon+}$) (Zitzler et al., 2003). These metrics effectively capture both the approximate sets' convergence and diversity. I_{HV} measures both convergence and diversity of a Pareto approximate front. It is the ratio of the Lebesgue measure of the objective space between a Pareto approximate front and a Pareto reference front. I_{GD} measures the average Euclidean distance in the objective space between each solution point on a Pareto reference front and its closest solution point on a Pareto approximate front. $I_{\varepsilon+}$ evaluates the minimum distance a Pareto approximate front must be shifted to dominate the best-known Pareto front in the objective space, which measures the convergence and consistency of a solution set.

In order to report any significant difference in the above metrics across different MOEAs, a nonparametric analysis was implemented in this chapter (Hadka and Reed, 2012; Ameca-Alducin et al., 2018). The one-way Kruskal-Wallis test (Kruskal and Wallis, 1952), with Dunn's D post-test (Dunn, 1964), was used to evaluate if a pair of end of run metric data sets differ significantly from each other. In this chapter, the metric data sets for an MOEA for represent an end of run metrics (I_{GD}, I_{HV}, or $I_{\varepsilon+}$) of the approximate Pareto sets for 30 duplicated runs. The score panel is used to evaluate the effectiveness of each group of data. The two groups' data is assigned as equivalent, if the difference is not significant (p-value>5%). In this condition, two groups are assigned using a zero score. Otherwise, the group with a better median value (i.e. lower values for I_{GD} and $I_{\varepsilon+}$ and higher value for I_{HV}) of a metric is recognised as the better performing group by having one score; and the other is penalised by assigning a zero score to it. There are overall 210 pairwise statistical tests for each end of run metric, which were conducted for all algorithm group pairs, and the scores for each algorithm group are added together to represent the effectiveness of performance for each algorithm. Therefore, the ranks for each MOEA metric can be sorted, based on the score for each group of data.

4.2.6.2. Percentage contribution to new best-known Pareto fronts

To compare the proposed MOEA with other state-of-the-art MOEA results for the case studies using a full computational budget, the contribution percentage, proposed by Wang et al. (2015), was used to evaluate how many solutions of the aggregate sets contribute to the new-best-known fronts for each case study. The details for calculating the contribution percentage for a case study are outlined as follows: firstly, all approximate fronts' objective values are rounded to the required epsilon precisions (EF_c and EI_n), as defined by Wang et al., (2015) for each case study. Then, the aggregated sets of the 30 duplicated runs are merged and yield a unique nondominated solution set. Thus, the contribution percentage of the unique nondominated set to the new-best-known Pareto fronts can be calculated. It is noted that the new best-known Pareto fronts for the case studies have been updated by the proposed MOEA, based on the best-known Pareto fronts updated by Jahanpour et al., (2018).

4.3. Results and Discussion

4.3.1. Performance of MOEAs for a Partial Computational Budget

The rankings of the constructed GMOEA algorithms based on the three end-of-run metrics' average are shown in Table 4-4. The first column consists of the name of each GMOEA, which is identified by the component options as described in Section 4.2.5, where the naming convention is: GMOEA(*A*, *B*, *C*, *D*) where *A* is the selection strategy (either the parent selection or survivor selection), *B* is the number of operators, *C* is the selection rule, and *D* is the metadata. For example, an algorithm with a crowding distance selection strategy; 12 operators; transition probability as the selection rule; and the archive as the metadata is denoted as GMOEA (CD, 12, T, A).

Table 4-4. Ranks of the 20 Constructed MOEAs

Algorithm	Ranking			
	GD	HV	E+	Average
GMOEA(CHC_{gen},6,T,A)	5	5	6	5.3
GMOEA(CHC_{gen},6,T,P)	7	5	10	7.3
GMOEA(CHC_{gen},6,S,A)	8	14	17	13
GMOEA(CHC_{gen},6,S,P)	10	13	15	12.7
GMOEA(CHC_{gen},6,N)	12	7	9	9.3
GMOEA(CD,6,T, A)	16	16	14	15.3
GMOEA(CD,6,TP)	17	18	16	17
GMOEA(CD,6,S,A)	19	19	19	19
GMOEA(CD,6,S,P)	19	20	20	19.7
GMOEA(CD,6,N)	18	11	7	12
GMOEA(CHC_{gen},12,T,A)	**2**	**1**	**1**	**1.3**
GMOEA(CHC_{gen},12,T,P)	1	2	2	1.7
GMOEA(CHC_{gen},12,S,A)	4	8	3	5
GMOEA(CHC_{gen},12,S,P)	6	12	11	9.7
GMOEA(CHC_{gen},12,N)	3	3	3	3
GMOEA(CD,12,T,A)	10	10	12	10.7
GMOEA(CD,12,T,P)	9	9	5	7.7
GMOEA(CD,12,S,A)	13	15	13	13.7
GMOEA(CD,12,S,P)	14	17	18	16.3
GMOEA(CD,12,N)	15	3	8	8.7

As can be seen, GMOEA(CHC_{gen},12,T,A) achieves the best performance, indicated by the overall average rank (Table 4-4). In contrast, GMOEA(CD, 6, S, P) achieves the overall worst performance of the three metrics ranks. It is obvious that different component combinations have an impact on algorithm performance, as shown by the average rank. Moreover, the table shows which algorithms contain a certain component that typically affects performance. For example, algorithms with 12 operators typically outperform algorithms with 6 operators, shown by the average rank (Table 4-3). To further investigate how each component's alternative affects MOEA performance, as outlined in Table 4-5, the aggregated average ranks of the algorithms using the same component are discussed as follows:

Table 4-5. The Average Rank of Each Component

Selection Strategy		Number of Operators		Hyperheuristic			
				Metadata		Selection Rule	
CHC_{gen}	6.7	**6**	13.2	**A**	10.6	**T**	8.3
CD	14.3	**12**	7.8	**P**	11.6	**S**	14.0
						N	8.0

As can be seen in Table 4-5, it is demonstrated that the selection strategy and the number of operators have greater impact on algorithm performance than other criteria. For example, the average ranks of selection strategy, CHC_{gen}(6.7) is over two times better than the CD (14.3). This result is consistent with the finding shown in (Wang et al., 2020b). The feature of convex hull contribution would avoid extensively sampling from solutions on the extended tails of the Pareto front, i.e., near vertical or near horizontal lines in the biobjective (Asadzadeh et al. 2014). This feature improves the algorithm's efficiency of searching. On the other hand, for the operator component, 12 operators (7.8) are about two times better than 6 operators (13.2). The finding is consistence with the results shown by Wang et al. (2020). For an algorithm that includes more operators, a higher search diversity enabled. Thus, performance should be improved (Wang et al. 2020).

For the influence of the metadata of a hyperheuristic, the average rank of the archive (A) (10.6) is slightly better than the population (P) (11.6) (Table 4-5). The ε-nondominated solutions are able to maintain convergence and diversity simultaneously (Hadka and Reed, 2013). Hence, an operator that can produce offspring with great convergence and diversity would benefit the search (Maier et al. 2014). This implies the ε-nondominated solution metadata from the archive could provide more effective feedback for operator prioritisation than the population.

For the selection rule, the average rank of the transitional selection rule (T) is 5.8 lower than the average rank of the steady selection rule (S) (Table 4-5), which suggests the transitional selection rule improves algorithm performance. The inferred reason for this is that transitional selection allows the all operators to produce offspring solutions, which increase the search diversity (Wang et al. 2020). In contrast, the steady selection rule only allows one operator to be selected and produce offspring solutions, which does not help to maximize search diversity. The naïve selection rule performs significantly better than the steady selection rule (Table 4-5). The reason is that naïve selection allows each operator to produce offspring solutions as well. However, it is unexpected to see the naïve average rank is marginally better than the transitional selection rule. Nonetheless, as shown in Table 4-3, the algorithm with a naïve selection rule does not always outperform the algorithm with the transitional selection rule. The results are discussed in Section 4.3.2.

4.3.2. Pairwise Analysis of Component Influences

To understand the relative influence of each pairwise combination of the different process components, Table 4-6 shows all pairwise combinations' merged average rank (across all algorithm instances).

Table 4-6. The Relative Influence of Each Pairwise Option of the Search Strategies

Primary Component	**Secondary Component**					
	Metadata		**Selection Rule**		**Selection Strategy**	
	Combination	**Average Rank**	**Combination**	**Average Rank**	**Combination**	**Average Rank**
Number of operators	**6-A** **6-P**	13.5 14.3	**6-T** **6-S** **6-N**	11.3 16.5 10.5	**6-CHC_{gen}** **6-CD**	9.4 17
	12-A **12-P**	7.8 9	**12-T** **12-S** **12-N**	5.3 11.5 5.5	**12-CHC_{gen}** **12-CD**	4 11.6
Selection Strategy	**CHC_{gen}-A** **CHC_{gen}-P**	6 7.8	**CHC_{gen}-T** **CHC_{gen}-S** **CHC_{gen}-N**	3.5 10.3 6		
	CD-A **CD-P**	15.3 15.5	**CD-T** **CD-S** **CD-N**	13 17.8 10		
Selection Rule	**T-A** **T-P**	8.3 13				
	S-A **S-P**	8.3 15				

As can be seen, the performance of each component in each pairwise component is typically the same as the results shown on Table 4-5. For example, for algorithms with the same number of operators, the Archive metadata (e.g. 12-A average rank is 7.8) is better than the Population metadata (e.g. 12-P average rank is 9). For algorithms with the same number of operators, the CHC_{gen} selection strategy (e.g. 12- CHC_{gen} average rank is 4) is more than two times better than the CD selection strategy (e.g. 12-CD average rank is 11.6). Moreover, algorithms with 12 operators are typically better than algorithms with 6 operators (Table 4-6). Thus, it is implied that the relative influence of the number of operators is greater than that of the selection strategy and greater than that of the metadata.

However, an exception was observed for the relative influence of the selection rules under different selection strategy options, as shown in Table 4-6. For example, the average rank of CD-N (10) is better than CD-T (13). These ranks resulted in the average rank of the transitional selection rule (8.3) being slightly worse than the average rank of the naïve selection rule (8.0) shown in

Table 4-5. However, algorithms with the CHC_{gen} selection strategy and transitional selection rule (CHC_{gen} -T) perform better than those with the naïve selection rule (CHC_{gen} -N). This implies an ineffective selection strategy (CD) could encourage non-effective operators to explore the search space and result in poor performance.

In summary, by conducting a comprehensive numerical study and analysing the relative influence of each component's impact on performance, typically, there is little correlation across each pairwise component. The relative influence of a component on algorithm performance would be the same when paired with components of other MOEA processes. This implies that the components in GMOEA(CHC_{gen},12,T,A) would improve other algorithms.

4.3.3. Benchmarking with Existing Algorithms

To evaluate and compare the performance of the GMOEA(CHC_{gen},12,T,A), with state-of-the-art MOEAs, the percentage contribution of solutions to the best known-Pareto fronts of the six case studies that were produced by the eight MOEAs (NSGA-II, ε-MOEA, ε-NSGA-II, AMALGAM, Borg, PADDS-CHC, PADDS-HVC) and GMOEA(CHC_{gen},12,T,A) are shown in Table 4-7. It can be seen that GMOEA(CHC_{gen},12,T,A) achieves the highest percentage contribution on the medium and large scale problems. Moreover, GMOEA(CHC_{gen},12,T,A) identified at least more than 44% of the new best-known Pareto front solutions for the large scale problem. In particular, for the largest scale problem, BIN, GMOEA(CHC_{gen}, 12, T, A) almost provides an entire new best-known Pareto front (267 out of 270), with reference to this phenomenon reported in Jahanpour et al. (2018). This means that GMOEA(CHC_{gen}, 12, T, A) is a highly effective MOEA to solve complicated WDS optimisation problems.

However, for the small scale problems, PADDS algorithms results in the highest percentage. For instance, the PADDS-HVC percentage contribution is 100 and 97.4 and GMOEA(CHC_{gen},12,T,A) has a percentage contribution of 53.1 and 66.7 for the NYT and HAN problems, respectively. According to the No-free-lunch theorem, it is not possible to have an algorithm that outperforms all other algorithms on every single problem in terms of solution quality and efficiency. Consequently, despite the relative simplicity of PADDS-HVC in comparison with GMOEA(CHC_{gen},12,T,A), this simpler algorithm performs more effectively for smaller problems, but cannot match the performance of GMOEA(CHC_{gen},12,T,A) for larger problems. Similarly, it is noted that for NSGA-II, the percentage contribution values are higher than other state-of-the-art MOEAs (i.e. AMALGAM, Borg, GMOEA(CHC_{gen},12,T,A)) for smaller problems. This fact implies that an MOEA with a simple structure can potentially achieve a more effective performance on small scale problems.

Despite the fact that GMOEA(CHC_{gen},12,T,A) was not as effective on the small scale problems, its performance dominates the existing MOEAs on medium and large scale problems, as indicated by it finding the greatest number of the best-known Pareto front solutions.

Table 4-7. The Percentage of Contribution to the Best-Known Pareto Front for Each MOEA

Problem	Number of new PF found by GMOEA-(CHC_{gen},12,T,A)	Number of solutions in best-known PF	Percentage contribution (%)							
			NSGA-II	ε-MOEA	ε-NSGA-II	AMALGAM	Borg	PADDS-CHC	PADDS-HVC	GMOEA-(CHC_{gen},12, T,A)
NYT	0	145	91.0	17.9	24.8	90.3	20.0	**100.0**	**100.0**	53.1
HAN	0	39	94.9	20.5	23.1	84.6	25.6	94.9	**97.4**	66.7
FOS	101	131	6.1	0.0	0.0	0.7	0.0	0.0	0.0	**95.2**
PES	79	119	4.0	11.6	0.0	1.3	0.0	20.1	25.6	**64.3**
MOD	90	201	15.9	0.0	0.0	11.1	0.0	29.5	0.0	**49.3**
BIN	267	270	0.0	0.0	0.0	0.4	0.0	1.1	0.0	**98.9**

Note: The highest percentage numbers for each case study are in bold.

4.4. Conclusion

This chapter investigates the influence of each process of MOEA on algorithm performance, and develops a new GMOEA(CHC_{gen},12,T,A), for solving WDS problems. Specifically, the objective of this chapter is that within a common MOEA framework, a systematic numerical experiment is conducted to investigate the impact of component combinations on algorithm performance. In the second objective, the best performing component combination MOEA is selected and compared with seven existing MOEAs on WDS problems.

To study the impact of each individual and pairwise component on algorithm performance, the GMOEA framework (Wang et al. 2020) was used to enable swapping of components of each algorithm process. There are eight components in the processes of reproduction (number of operators, hyperheuristic) and selection (including parent selection and survivor selection) that together form 20 constructed algorithms. A comprehensive numerical experiment was conducted to study the relative impact of each individual and pairwise component on algorithm performance. Three end of run metrics with rigorous statistical tests were used to assess algorithm performance.

The results indicate each component affects algorithm performance. The number of operators and selection strategy play important roles in affecting the performance. The reason is that they affect the population solutions directly by controlling search behaviour. Moreover, it is found that a high number of operators would improve algorithm performance; and the CHC_{gen} selection strategy is more effective than the CD selection strategy. These findings are the same as in previous studies (Wang et al. 2020a & 2020b). In addition, it is found that the influence of each component on algorithm performance is typically independent from other components.

The proposed new algorithm, GMOEA(CHC_{gen},12,T,A), was developed by selecting the most effective one out of the 20 constructed algorithms. The performance of GMOEA(CHC_{gen},12,T,A) was assessed by evaluating the contribution to the best-known Pareto Fronts from the six WDS problems and comparing them with seven existing MOEAs (NSGA-II, ε-MOEA, ε-NSGA-II, AMALGAM, Borg, PADDS-CHC and PADDS-HVC). GMOEA(CHC_{gen},12,T,A) outperformed the existing MOEAs by finding many new best-known Pareto Front solutions. In particular, for the most complex BIN problem, GMOEA(CHC_{gen},12,T,A) found 98.9% new best-known Pareto Front solutions. However, for the small scale problems, a simple structure algorithm like NSGA-II outperforms GMOEA(CHC_{gen},12,T,A). Hence, for future study, it is possible to investigate the relationship between the complexity of a problem type and algorithm structure. Alternatively, it would be possible to design an algorithm with an adaptive algorithm structure.

Chapter 5 Conclusions

5.1. Research Contributions

MOEAs have the ability to adjust the way they search through the solution spaces by either intensifying the search in promising regions or diversifying the search in less promising regions, enabling them to perform well on problems with different characteristics, as the search behaviour alterations can be fine-tuned by the components of each process of the MOEA. This thesis has investigated the impact of the components of MOEAs' search behaviour and performance comprehensively. The details of the findings are outlined as follows:

The impact of the operator set size on the performance of multi-objective evolutionary algorithms (MOEAs) for WDS problems has been studied comprehensively (Chapter 2). The study assessed (i) the relative influence of the size of the operator set on algorithm performance, (ii) whether the size of the operator set is more important than the composition of the operator set, (iii) whether the size of the operator set is more important than the combined effect of the composition of the operator set and the search strategies used, and (iv) the potential for improving the performance of existing MOEAs by increasing the size of the operator set.

The results from the 3,150 optimisation runs for the work presented in Chapter 2 clearly indicate that operator set size plays a dominant role in algorithm performance. Operator set size was observed to have a larger influence than operator parameter values, operator set composition and other strategies affecting algorithm searching behaviour. The reason for the increased performance of algorithms using a larger number of operators is that they provide a greater variety of searching mechanisms, which are able to find better solutions at different stages of the optimisation process.

Overall, the findings within Chapter 2 tend to suggest that existing multi-objective evolutionary algorithms do not use a sufficient number of operators and that there is significant potential to increase the performance of a wide range of existing algorithms simply by increasing their operator set size. Based on the results obtained, it is recommended that the number of operators in existing algorithms should be increased to between 10 and 12, ensuring a balance between exploration and exploitation. For cases where the original algorithm to be improved does not use a hyperheuristic to control the degree to which each operator contributes to the search at each iteration, it is recommended to use the NAÏVE hyperheuristic, which ensures that all operators contribute equally.

A novel selection strategy called the generational convex hull contribution is proposed for generational MOEAs (GMOEAs) (Chapter 3). Moreover, the impact of selection strategies on algorithm performance has been investigated by comparing the performance of GMOEAs that use different existing selection strategies.

As shown in Chapter 3, the general MOEA framework algorithm with the CHC_{Gen} selection strategy outperforms the four other existing selection strategies that were tested. The CHC_{Gen} selection strategy not only accounts for convergence and diversity in generating the approximate front, but its search behaviour is well-suited for problems that have a convex shape of the approximate fronts. Moreover, the selection strategy should result in better solutions being identified within the 'knee regions' of approximate Pareto fronts.

CHC_{Gen} was compared with four existing selection strategies by implementing these strategies within a consistent general MOEA framework. The general MOEA framework with the CHC_{Gen} selection strategy was found to outperform four other popular existing selection strategies in the numerical study involving six WDS problems. The CHC_{Gen} selection strategy showed the best overall convergence, diversity and consistency of the approximate fronts that were generated.

As shown in Chapter 3, the CHC_{Gen} selection strategy augments the selection preference to bias the population solutions that lie on the convex hull regions of the approximate front. Given the nature of the convex hull solutions within a non-dominated set that are closer to the “ideal point” and in distinct regions along the approximate front, this type of selection preference leads to an improved convergence and diversity of the search. Therefore, the CHC_{Gen} selection strategy allows the algorithm to explore the search space effectively and results in the best performance of the approximate front in comparison with other existing selection strategies.

To further investigate the potential of the CHC_{Gen} selection strategy to improve existing MOEAs, the current best generational MOEA for solving WDS problems (GALAXY) and the industry standard generational MOEA (NSGA-II) were modified to incorporate the CHC_{Gen} selection strategy. The CHC_{Gen} selection strategy was found to be able to boost the performance of these two algorithms, suggesting that the proposed selection strategy could benefit other existing MOEAs.

A systematic numerical experiment was conducted to investigate the impact of component combinations on algorithm performance (Chapter 4). Furthermore, a new MOEA has been developed by combining the most effective components considered in this thesis.

The results from Chapter 4 indicate that each component affects algorithm performance. The number of operators and the selection strategy play important roles in affecting performance. The reason is that they affect the population of solutions directly by controlling the search behaviour. Moreover, it was found that a high number of operators would improve algorithm performance and that the CHC_{gen} selection strategy is more effective than the CD selection strategy. These findings are the same as in previous studies (Wang et al. 2020a & 2020b). In addition, it was found that the influence of each component on algorithm performance is typically independent from other components.

Within Chapter 4, the proposed new algorithm, GMOEA(CHC_{gen},12,T,A), was developed by selecting the most effective artificially constructed algorithm out of a total of 20 algorithms. The performance of GMOEA(CHC_{gen},12,T,A) was assessed by evaluating its contribution to the best-known Pareto Fronts from the six WDS problems and comparing them with seven existing MOEAs (NSGA-II, ε-MOEA, ε-NSGA-II, AMALGAM, Borg, PADDS-CHC and PADDS-HVC). The algorithm GMOEA(CHC_{gen},12,T,A) outperformed the existing MOEAs by finding many new best-known Pareto front solutions. In particular, for the most complex case study, the BIN problem, GMOEA(CHC_{gen},12,T,A) found 98.9% of the new best-known Pareto front solutions. However, for the small scale problems, a simply structured algorithm, like NSGA-II, outperforms GMOEA(CHC_{gen},12,T,A).

5.2. Scope of Future Work

The recommendation for future work related to multi-objective evolutionary algorithms are outlined below.

The proposed GMOEA has been developed and tested in WDS problems. The new challenge is to test the effectiveness of the proposed GMOEA on other optimisation problems. For example, it is worth evaluating different water resource problems such as hydraulic model calibration problems, water quality optimisation problems, etc. Moreover, this thesis only considers a bi-objective optimisation problem. For real-world WDS problems, more design objectives should be considered, such as not only the economic perspective, but also the community, performance and environmental perspectives. Considering multiple objectives enables us not only to explore the algorithm's performance in higher objective dimensions, but is also more practical for designing WDSs in the real world.

One of the key advantages of MOEAs is the ability to adjust the way they search through the solution spaces by either intensifying the search in promising regions (i.e., exploiting good solutions) or diversifying the search in less promising regions (e.g., exploring the solutions space more widely), enabling them to perform well for problems with different problem characteristics. However, the relationship between the problem characteristics and algorithm search behaviour is still not sufficiently clear. Bridging this gap would help to develop a better understanding of the impact of search behaviour on algorithm performance. Hence, it would be possible to tune the algorithm search behaviour according to problem characteristics, thereby significantly improving performance.

This thesis views MOEAs as a generic process. Through adopting different search strategies, the search behaviour could be changed, thereby affecting algorithm performance. However, these search strategies are predetermined prior to the search. In future work, there is great potential to adapt different search strategies during the search, which allows the algorithm to show a more flexible search behaviour and search diversity. To achieve this, another challenge is to propose a robust feedback to collect the problem characteristics during the search and convert them to useful information (e.g., desired search behaviour) to reward the most effective search strategies for conducting the search.

The limitation of deterioration, in which good solutions are erroneously replaced by worse solutions, has existed in generational MOEAs. This is due to the fact that the size of a population in generational MOEAs is fixed. The challenge is to develop a method to preserve all elite solutions in the population and retain some space to store redundant solutions to increase search diversity. Also, the effectiveness of such a method should be proved mathematically. If the deterioration can be solved for generational MOEAs, it would improve the algorithms' performance.

This thesis only investigates the potential to swap different search strategies on generational MOEAs. The conclusions found are not tested on the other type of MOEA, which is a steady state MOEA. Steady state MOEAs have also been developed and applied to a wide range of optimisation problems for over two decades. It would be worthwhile testing the potential of swapping different search strategies to improve algorithm performance. By doing this, more generic conclusions could be determined.

References

Ameca-Alducin, M. Y., Hasani-Shoreh, M., Blaikie, W., Neumann, F., & Mezura-Montes, E. (2018). A comparison of constraint handling techniques for dynamic constrained optimization problems, In 2018 IEEE Congress on Evolutionary Computation, CEC 2018.

Asadzadeh, M., & B. A. Tolson (2013). Pareto archived dynamically dimensioned search with hypervolume-based selection for multiobjective optimization Engineering Optimization, 45(12), 1489–1509.

Asadzadeh, M., Tolson, B. A., & Burn, D. H. (2014). A new selection metric for multiobjective hydrologic model calibration. Water Resources Research, 51, 7082–7099.

Back, T. (1996). Evolutionary Algorithms in Theory and Practice. Oxford University Press, New York.

Barber, C. B., Dobkin, D. P., & Huhdanpaa, H. (1996). The quickhull algorithm for convex hulls. ACM Transactions on Mathematical Software, 22(4), 469–483.

Bader, J., & E. Zitzler. (2011). Hype: An Algorithm for Fast Hypervolume-Based Many-Objective Optimization. Evolutionary Computation, 19, 45–76.

Bi, W., Maier, H. R., & Dandy, G. C. (2016). Impact of starting position and searching mechanism on evolutionary algorithm convergence rate. Journal of Water Resources Planning and Management, 142(9).

Burke, E., Kendall, G., Newall, J., Hart, E., Ross, P., & Schulenburg, S. (2002). Hyper-Heuristics: An emerging direction in modern search technology, In Handbook of Metaheuristics (pp. 457–474).

Burke, E. K., Gendreau, M., Hyde, M., Kendall, G., Ochoa, G., Özcan, E., & Qu, R. (2013). Hyper-heuristics: A survey of the state of the art. Journal of the Operational Research Society, 64(12), 1695–1724.

Cheung, P. B., Reis, L. F. R., Formiga, K. T. M., Chaudhry, F. H., & Ticona, W. G. C. (2003). Multiobjective evolutionary algorithms applied to the rehabilitation of a water distribution system: A comparative study. Proceedings, 2nd International Conference on Evolutionary Multi-Criterion Optimization, Springer, Faro, Portugal.

Deb, K., & Agrawal, R. B. (1994). Simulated binary crossover for continuous search space. Technical Report IITK/ME/SMD-94027, Indian.

Deb, K., Joshi, D., & Anand, A. (2002a). Real-coded evolutionary algorithms with parent-centric recombination. In Proceedings of the World Congress on Computational Intelligence pp. 61–66.

Deb, K., Pratap, A., Agarwal, S., & Meyarivan, T. (2002b). A Fast and elitist multiobjective genetic algorithm: NSGA-II. IEEE Transactions on Evolutionary Computation, 6(2), 182–197.

Deb, K., M. Mohan, & S. Mishra (2005). Evaluating the e-domination based multi-objective evolutionary algorithm for a quick computation of pareto-optimal solutions. Evolutionary Computation, 13(4), 501–525.

Drake, J. H., Kheiri, A., Özcan, E., & Burke, E. K. (2019). Recent advances in selection hyper-heuristics. European Journal of Operational Research, 274 (2), 405-428.

Dunn, O. J. (1964). Multiple comparisons using ranks sums. Technometrics, 6, 241–252.

Eiben, A. E., and Smith, J. E. (2015). Introduction to Evolutionary Computing. Heidelberg: Springer.

Emmerich M., Beume N., Naujoks B. (2005). An EMO Algorithm Using the Hypervolume Measure as Selection Criterion. Evolutionary Multi-Criterion Optimization EMO 2005, Lecture Notes in Computer Science, vol 3410, Coello Coello C.A., Hernández Aguirre A., Zitzler E., eds., Springer, Berlin, Heidelberg, 62–76.

Engelhardt, M. O., Skipworth, P. J., Savic, D. A., Saul, A. J., & Walters, G. A. (2000). Rehabilitation strategies for water distribution networks: A literature review with a UK perspective. Urban Water, 2(2), 153-170.

Farmani, R., Savic, D. A., & Walters, G. A. (2004). EXNET benchmark problem for multi-objective optimization of large water systems. Modelling and Control for Participatory Planning and Managing Water Systems. IFAC Workshop, Venice, Italy.

Fenicia, F., Savenije, H. H. G., Matgen, P. & Pfister, L. (2007). A comparison of alternative multiobjective calibration strategies for hydrological modelling. Water Resources Research. 43(3).

Fu, G., Kapelan, Z., Kasprzyk, J., & Reed, P. (2012a). Optimal design of water distribution systems using many-objective visual analytics. Journal of Water Resources Planning and Management, 624-633.

Haario, H., Saksman, E., & Tamminen, J. (2001). An adaptive metropolis algorithm, Bernoulli. 7(2), 223-242

Hadka, D., & Reed, P. (2012). Diagnostic assessment of search controls and failure modes in many-objective evolutionary optimization. Evolutionary Computation, 20(3), 423–452.

Hadka, D., & Reed, P. (2013). “Borg: An auto-adaptive many-objective evolutionary computing framework.” Evolutionary Computation, 21(2), 231–259.

Hadka, D., & Reed, P. (2015). Large-scale parallelization of the Borg multiobjective evolutionary algorithm to enhance the management of complex environmental systems. Environmental Modelling and Software, 69, 353–369.

Hanne, T. (1999). On the convergence of multiobjective evolutionary algorithms. European Journal of Operational Research, 117(3), 553–564.

Igel, C., & Hansen, N. (2007). Covariance Matrix Adaptation for Multi-objective Optimization. Evolutionary Computation, 15(1), 1–28.

Jahanpour, M., Tolson, B. A., & Mai, J. (2018). PADDS algorithm assessment for bi-objective water distribution system benchmark design problems. Journal of Water Resources Planning and Management, 144(3), 1–12.

Jourdan, L., Corne, D. W., Savic, D., & Walters, G. (2005). LEMMO: hybridising rule Induction and NSGA-II for multi-objective water systems design. CCWI.

Karafotias, G., Hoogendoorn, M., & Eiben, A. E. (2015). Parameter control in evolutionary algorithms: Trends and challenges. IEEE Transactions on Evolutionary Computation, 19(2), 167–187.

Kennedy, J., & Eberhart, R. (1995), Particle swarm optimization. In Proceedings of IEEE International Conference on Neural Networks pp. 1942–1948, Perth, Australia.

Khu, S. T. & Keedwell, E. (2005). Introducing more choices (flexibility) in the upgrading of water distribution networks: the New York city tunnel network example. Engineering Optimization, 37(3), 291-305.

Kita, H., Ono, I., & Kobayashi, S. (1999). Multi-parental extension of the unimodal normal distribution crossover for real-coded genetic algorithms. In Proceedings of Congress on Evolutionary Computation pp. 1581–1588.

Knowles, J. D., D. W. Corne, & M. Fleischer. (2003). Bounded Archiving Using the Lebesgue Measure. Proceedings of the IEEE Congress on Evolutionary Computation, New York, 2490–2497.

Kollat, J. B., & Reed P. M. (2005). The value of online adaptive search: A performance comparison of NSGA-II, ε-NSGAII, and εMOEA, paper presented at the 3rd International Conference on Evolutionary Multi-Criterion Optimization, Springer, Berlin, Heidelberg, Guanajuato, Mexico, 9-11 March.

Kollat, J. B., & Reed, P. M. (2007). A computational scaling analysis of multiobjective evolutionary algorithms in long-term groundwater monitoring applications. Advances in Water Resources, 30(3), 408–419.

Kruskal, W., & Wallis, A. (1952). Use of ranks in one-criterion variance analysis. Journal of the American Statistical Association, 47(260), 583–621.

Maier, H. R., Kapelan, Z., Kasprzyk, J., Kollat, J., Matott, L. S., Cunha, M. C., et al. (2014). Evolutionary algorithms and other metaheuristics in water resources: current status, research challenges and future directions. Environmental Modelling & Software, 62, 271–299.

Maier, H. R., Razavi, S., Kapelan, Z., Matott, L. S., Kasprzyk, J., & Tolson, B. A. (2019). Introductory overview: Optimization using evolutionary algorithms and other metaheuristics. Environmental Modelling and Software, 114, 195–213.

Mala-Jetmarova, H., Sultanova, N., & Savic, D. (2017). Lost in optimisation of water distribution systems? A literature review of system operation. Environmental Modelling & Software, 93, 209–254.

Mala-Jetmarova, H., Sultanova, N., & Savic, D. (2018). Lost in optimisation of water distribution systems? A literature review of system design. Water, 10(3), 307.

Michalewicz, Z. (1992). Genetic algorithms + data structures = evolution programs. Springer-Verlag, New York.

Munoz, M. A., & Smith-Miles, K. A. (2017). Performance analysis of continuous black-box optimization algorithms via footprints in instance space. Evolutionary Computation, 25(1), 529–554.

Murtagh, B. A., & Saunders, M. A. (1987). MINOS 5.1 User's Guide. Systems Optimization Laboratory, Dept. of Operations Research, Stanford University, Stanford, Calif.

Nicklow, J., Reed, P., & Savic, D. (2010). State of the art for genetic algorithms and beyond in water resources planning and management. Journal of Water Resources Planning and Management, 136(4), 412–432.

Prasad, T. D., & Park, N.-S. (2004). Multiobjective genetic algorithms for design of water distribution networks. Journal of Water Resources Planning and Management, 130(1), 73-82.

Pulido, G. T. & Coello Coello, C. A. (2004). A constraint-handling mechanism for particle swarm optimization. In Proceedings of the 2004 Congress on Evolutionary Computation, pp. 1396-1403.

Rechenberg, I. (1965). Cybernetic solution path of an experimental problem, Royal Aircraft Establishment, Library Translation, 1122, Farnborough, England.

Reed, P. M., Hadka, D., Herman, J. D., Kasprzyk, J. R., & Kollat, J. B. (2013). Evolutionary multiobjective optimization in water resources: The past, present, and future. Advances in Water Resources, 51, 438–456.

Rossman, L. A. (2000). EPANET 2 users manual, technical report, EPA/600/R-00/057. Washington, DC: U.S. Environmental Protection Agency.

Rudolph, G. (1999). Evolutionary Search Under Partially Ordered Fitness Sets. Technical Report, CI-67/99.

Soria-Alcaraz, J. A., Ochoa, G., Sotelo-Figeroa, M. A., & Burke, E. K. (2017). A methodology for determining an effective subset of heuristics in selection hyper-heuristics. European Journal of Operational Research, 260, 972-983.

Schaake J. & Lai D., Linear programming and dynamic programming applications to water distribution network design. Dept. Civil Eng. Mass. Inst. Technol., Cambridge, MA, Res. Rep. 116, 1969.

Simpson, B. A. R., & Dandy, G. C. (1994). Genetic algorithms compared to other techniques for pipe optimization. Journal of Water Resources Planning and Management, 120(4), 423–443.

Singh, A., Minsker, B., & Goldberg, D. E. (2003). Combining reliability and pareto optimality – an approach using stochastic multi-objective genetic algorithms. World Water and Environmental Resources Congress 2003, June 23-26 2003, American Society of Civil Engineers, Philadelphia, PA, US.

Storn, R. & Price, K. (1997). Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. Journal of Global Optimization." 11(4), 341-359.

Szemis, J. M., Dandy, G. C., & Maier, H. R. (2013). A multiobjective ant colony optimization approach for scheduling environmental flow management alternatives with application to the River Murray, Australia. Water Resources Research, 49(10), 6399–6411.

Tsutsui, S., Yamamura, M., & Higuchi, T. (1999). "Multi-parent recombination with simplex crossover in real coded genetic algorithms." In Proceedings of the 1999 Genetic and Evolutionary Computation Conference (GECCO'99) pp. 657-664.

Veldhuizen, D. A. V. (1999). Multiobjective evolutionary algorithms: Classifications, analyses, and new innovations, Doctoral thesis, 272 pp., Air Force Institute of Technology, Wright-Patterson Air Force Base, O H.

Vrugt, J. a, & Robinson, B. a. (2007). Improved evolutionary optimization from genetically adaptive multimethod search. In Proceedings of the National Academy of Sciences of the United States of America, Vol. 104, pp. 708-711.

Vrugt, J. A., Robinson, B. A., & Hyman, J. M. (2009). Self-adaptive multimethod search for global optimization in real-parameter spaces. IEEE Transactions on Evolutionary Computation, 13(2), 243-259.

Walski, T. M. (2001). The wrong paradigm, why water distribution optimization doesn't work. Journal of Water Resources Planning and Management, 203-205.

Wang, P., Zecchin, A. C., Maier, H. R., Zheng, F., & Newman, J. P. (2020a). Do existing multiobjective evolutionary algorithms use a sufficient number of operators? An empirical investigation for water distribution design problems. Water Resources Research, 55, e2019WR026031.

Wang, P., Zecchin, A. C., Maier, H. R., (2020b). A New Proposed Convex Hull Selection Strategy: Study on the Impact of Selection Strategy on Generational MOEA Performance for Water Distribution Design Problems. (Unpublished manuscript)

Wang, Q., Guidolin, M., Savic, D., & Kapelan, Z. (2015). Two-objective design of benchmark problems of a water distribution system via MOEAs: towards the best-known approximation of the true pareto front. Journal of Water Resources Planning and Management, 141(3).

Wang, Q., Savić, D. A., & Kapelan, Z. (2017). GALAXY: A new hybrid MOEA for the optimal design of water distribution systems. Journal of Water Resources Research, 53(3), 1-19.

Wolpert, D. H., & Macready, W. G. (1997). No free lunch theorems for search. IEEE Transactions on Evolutionary Computation, 1(1), 67-82.

Woodruff, M. J., Reed, P. M., and Simpson, T. W. (2013). Many objective visual analytics: Rethinking the design of complex engineered systems. Structural and Multidisciplinary Optimization, 48(1), 201–219.

Yu., X. & Gen, M. (2010). Introduction to Evolutionary Algorithms. Evolutionary Algorithms in Molecular Design. London, UK: Springer Verlag.

Zapotecas-Martinez, S., & Menchaca-Mendez, A. (2020). On the Performance of Generational and Steady-State MOEA/D in the Multi-Objective 0/1 Knapsack Problem. CEC 2020: 1–8.

Zecchin, A. C., Simpson, A. R., Maier, H. R., & Nixon, J. B. (2005). Parametric study for an ant algorithm applied to water distribution system optimization. IEEE Transactions on Evolutionary Computation, 9(2), 175-191.

Zecchin, A. C., Simpson, A. R., Maier, H. R., Marchi, A., & Nixon, J. B. (2012). Improved understanding of the searching behavior of ant colony optimization algorithms applied to the water distribution design problem. Journal of Water Resources Research, 48(9), 1-16.

Zeff, H. B., Herman, J. D., Reed, P. M., & Characklis, G. W. (2016). Cooperative drought adaptation: Integrating infrastructure development, conservation, and water transfers into adaptive policy pathways. Water Resources Research, 52, 7327-7346.

Zhang, X., Srinivasan, R., and Liew, M. Van. (2010). On the use of multi-algorithm, genetically adaptive multi-objective method for multi-site calibration of the SWAT model. Hydrological Processes, 24(8), 955–969.

Zheng, F., A. C. Zecchin, A. R. Simpson, & M. F. Lambert (2013). Noncrossover dither creeping mutation-based genetic algorithm for pipe network optimization. Journal of Water Resources Planning and Management. 140(4), 553–557

Zheng, F. (2015). Comparing the real-time searching behavior of four differential-evolution variants applied to water-distribution-network design optimization. Journal of Water Resources Planning and Management, 141(10), 04015016.

Zheng, F., Qi, Z., Bi, W., Zhang, T., Yu, T., & Shao, Y. (2017). Improved understanding on the searching behavior of NSGA-II operators using run-time measure metrics with application to water distribution system design problems. Water Resources Management, 31(4), 1121–1138.

Zheng, F., Simpson, A. R., & Zecchin, A. C. (2014). An efficient hybrid approach for multiobjective optimization of water distribution systems. Water Resources Research, 50, 3650–3671.

Zheng, F., Zecchin, A. C., Maier, H. R., & Simpson, A. R. (2016). Comparison of the searching behavior of NSGA-II, SAMODE, and Borg MOEAs applied to water distribution system design problems. Journal of Water Resources Planning and Management, 142(7).

Zheng, F., Zecchin, A. C., Newman, J. P., Maier, H. R., & Dandy, G. C. (2017). An adaptive convergence-trajectory controlled ant colony optimization algorithm with application to water distribution system design problems. IEEE Transactions on Evolutionary Computation, 21(5), 773–791.

Zheng, F., Zecchin, A. C., Simpson, A. R., & Lambert, M. F. (2013). Noncrossover dither creeping mutation-based genetic algorithm for pipe network optimization. Journal of Water Resources Planning and Management, 140(4), 553–557.

Zitzler, E., & Thiele, L. (1998). Multiobjective optimization using evolutionary algorithms—A comparative case study. In 5th International Conference on Parallel Problem Solving From Nature (PPSN V), Springer, Berlin, 292–301.

Zitzler, E., & Thiele, L. (1999). Multiobjective evolutionary algorithms: A comparative case study and the strength Pareto approach. IEEE Transactions on Evolutionary Computation, 3(4), 257–271.

Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., & Da Fonseca, V. G. (2003). Performance assessment of multiobjective optimizers: An analysis and review. IEEE Transactions on Evolutionary Computation, 7(2), 117–132.

Zitzler, E., Deb, K., & Thiele, L. (2000). Comparison of Multiobjective Evolutionary Algorithms: Empirical Results. Evolutionary Computation, 8(2), 173–195.

Zitzler, E., Thiele, L., Laumanns, M., Fonseca, C. M., and Da Fonseca, V. G. (2003). "Performance assessment of multiobjective optimizers: An analysis and review." IEEE Transactions on Evolutionary Computation, 7(2), 117–132.

Return the completed form to the
Adelaide Graduate Centre
graduate.centre@adelaide.edu.au

THE UNIVERSITY *of* ADELAIDE

Final Thesis Lodgement

Division of Research and Innovation

Following examination of your thesis, you are required to lodge one digital copy of your final thesis with the Graduate Centre, together with your completed Final Thesis Lodgement form. Subject to any restrictions on access that may apply, the digital copy will be included in the Adelaide Research & Scholarship (AR&S) Digital Repository and Trove.

On lodgement of your final copy, you and your School must certify that any and all required amendments have been satisfactorily completed and that any issues which may affect the digital publication of your thesis on the web have been addressed.

The following information will be made available to the University Library to facilitate the appropriate cataloguing of your thesis.

Personal Information

FAMILY NAME	WANG
OTHER NAMES	PENG
TITLE (Professor/Dr/Mr/Mrs/Ms/Miss/Mx)	Mr
STUDENT ID	a1660293
POSTAL ADDRESS	44a Seventh Ave, St Morris, SA 5068
EMAIL (Personal, Non University for future contact)	peng.w911101@gmail.com
SCHOOL/DISCIPLINE	School of Civil, Environmental and Mining Engineering
ACADEMIC PROGRAM (Full title e.g. Doctor of Philosophy)	Doctor of Philosophy

Document Information

TYPE (Conventional, Publication format etc)	Publication format
TITLE (as it appears on the title page of the thesis)	A Systematic Investigation of Multi-Objective Evolutionary Algorithms Applied to the Water Distribution System Problem
KEYWORDS (keywords or phrases separated by commas e.g. Ugh boots, footwear, foot apparel)	Multi-objective evolutionary algorithms, selection strategy, water distribution system design optimization, GALAXY, NSGA-II
QUALIFICATION YEAR (the year in which you completed the requirements for your award)	2021

Supervisor Information

PRINCIPAL SUPERVISOR	Dr Aaron C. Zecchin
ADDITIONAL SUPERVISORS NOTE: if not University of Adelaide staff please state organisation they are associated with.	Prof. Holger R. Maier

Page 1 of 2

Updated 14 August 2019

Certification of Student

FINAL THESIS

- ■ I hereby lodge one digital copy of my final thesis with the Adelaide Graduate Centre.
- ■ My thesis incorporates any and all required amendments.
- ■ I certify that the digital copy is a direct equivalent of the copy of my thesis approved by the University for the award of the degree. No emendation of content has occurred, and if there are any minor variations to formatting, they are a result of the conversion to Adobe Acrobat format.
- ■ I am aware that I am responsible for ensuring that the inclusion of the full text of the thesis in the AR&S Digital Repository will not jeopardise any future commercial publication plans.

COPYRIGHT STATEMENT

- ■ I hereby grant to the University of Adelaide or its agents the right to archive and to reproduce and communicate to the public my thesis or dissertation in whole or in part in the University libraries in all forms of media, now or hereafter known. I retain ownership of copyright and all other proprietary rights, such as patent rights. I also retain the right to use in future works (such as articles or books) all or part of this thesis or dissertation.

THIRD PARTY COPYRIGHT MATERIAL

- ☐ Section(s) of my thesis contain third party copyright material.

If ticked:

- ☐ I have obtained and attached copies of written permission from the copyright holder(s) to use some or all of this material. For assistance regarding copyright permissions contact the University's Copyright Coordinator: copyright@adelaide.edu.au
- ☐ I give permission for any third party copyright for which I have not obtained permission to be removed from the digital copy of my thesis. I have attached an errata sheet listing pages, sections and/or numbered figures that need to be removed for publication in the AR&S Digital Repository.

Peng Wang		27/06/2021
Student name (Please print)	Signature here	Date

Certification of School

SUPERVISOR CERTIFICATION

I have read and checked the final digital thesis and confirm that the candidate has made any required amendments to the satisfaction of the School.

Dr Aaron C. Zecchin		27/06/2021
Principal Supervisor's name (Please print)	Signature here	Date

HEAD OF SCHOOL OR POSTGRADUATE COORDINATOR CERTIFICATION

I approve the submission of the thesis for completion of the degree.

Scott Smith		05/07/2021
Head of School OR Postgraduate Coordinator's name (Please print)	Signature here	Date

Please note that if the Head is also one of the supervisor(s), the Postgraduate Coordinator must complete the certification. If the Postgraduate Coordinator is one of the supervisors, then the Head or Deputy Head must provide the certification. If both Head and Postgraduate Coordinator are supervisors, then the Deputy Head must provide the certification. If the Deputy Head is also a supervisor, the certification should be provided by another senior academic staff member.

Further Information

Ph: (08) 8313 5882
Fax: (08) 8313 5725
Email: graduate.centre@adelaide.edu.au
Web: www.adelaide.edu.au/graduatecentre

www.adelaide.edu.au/graduatecentre

Page 1 of 2

Updated 14 August 2019