

PUBLISHED VERSION

Vivian, R.; Falkner, K.; Falkner, N.

Addressing the challenges of a new digital technologies curriculum: MOOCs as a scalable solution for teacher professional development

Research in Learning Technology, 2014; 22(0): 24691-1- 24691-19

© 2014 R. Vivian et al. Research in Learning Technology is the journal of the Association for Learning Technology (ALT), a UK-based professional and scholarly society and membership organisation. ALT is registered charity number 1063519. <http://www.alt.ac.uk/>. This is an Open Access article distributed under the terms of the Creative Commons CC-BY 4.0 License (<http://creativecommons.org/licenses/by/4.0/>), allowing third parties to copy and redistribute the material in any medium or format and to remix, transform, and build upon the material for any purpose, even commercially, provided the original work is properly cited and states its license.

Originally Published at:

<http://dx.doi.org/10.3402/rlt.v22.24691>

PERMISSIONS

<http://researchinlearningtechnology.net/index.php/rlt/about/submissions#authorGuidelines>

Copyright and License Notice

Authors contributing to Research in Learning Technology agree to publish their articles under the terms of the Creative Commons CC-BY 4.0 License (<http://creativecommons.org/licenses/by/4.0/>) allowing third parties to copy and redistribute the material in any medium or format, and to remix, transform, and build upon the material, for any purpose, even commercially, under the condition that appropriate credit is given, that a link to the license is provided, and that you indicate if changes were made. You may do so in any reasonable manner, but not in any way that suggests the licensor endorses you or your use.

Please see [Copyright and Licence Agreement](#) for further details.

22 October 2014

<http://hdl.handle.net/2440/86483>

Addressing the challenges of a new digital technologies curriculum: MOOCs as a scalable solution for teacher professional development

Rebecca Vivian*, Katrina Falkner and Nickolas Falkner

School of Computer Science, The University of Adelaide, South Australia, Australia

(Received 17 April 2014; final version received 22 July 2014)

England and Australia have introduced new learning areas, teaching computer science to children from the first year of school. This is a significant milestone that also raises a number of big challenges: the preparation of teachers and the development of resources *at a national scale*. Curriculum change is not easy for teachers, in any context, and to ensure teachers are supported, scaled solutions are required. One educational approach that has gained traction for delivering content to large-scale audiences are massively open online courses (MOOCs); however, little is known about what constitutes effective MOOC design, particularly within professional development contexts. To prepare teachers in Australia, we decided to ride the wave of MOOCs, developing a MOOC to deliver free computing content and pedagogy to teachers with the integration of social media to support knowledge exchange and resource building. The MOOC was designed to meet teacher needs, allowing for flexibility, ad-hoc interactions, support and the open sharing of resources. In this paper, we describe the process of developing our initiative, participant engagement and experiences, so that others encountering similar changes and reforms may learn from our experience.

Keywords: scaling up; MOOCs; open access; professional development; online course; computer science education

Introduction

Despite the growing ubiquity of digital technologies (DT), university enrolments within information communication technologies (ICTs) remain relatively low (Koppi *et al.* 2013). National enrolment trends in Australia depict static enrolment figures since 2001 (Koppi *et al.* 2013) and more problematic is the low representation of females choosing to study computer science (CS), with 25% or less of ICT students being female (Koppi *et al.* 2013). A similar situation exists in the United States, with only 12.9% of females enrolled in undergraduate degrees (Zweben 2012). The root of such problems is often attributed to the perception of CS careers (Koppi *et al.* 2013), that CS is better suited to males and a career for ‘nerds’ or ‘geeks’ (Bock *et al.* 2013). To promote CS as an interesting career, global initiatives have been targeted at increasing youth engagement and interest in CS through various outreach programs (Bell *et al.* 2011; Koppi *et al.* 2013; Lambert & Guiffre, 2009; Liu *et al.* 2011; Mykietiak *et al.* 2012). However, the enrolment figures and research findings suggest that outreach programs have had little or no success (Koppi *et al.* 2013).

*Corresponding author. Email: rebecca.vivian@adelaide.edu.au

More recently, a drive to include computing in the school curriculum has arisen, proposing that all children should have an opportunity to develop computational thinking skills and have a chance to be ‘creators’ of DT (Gander *et al.* 2013; The Royal Society 2012). Little is known about what impact curriculum changes will have on enrolment and interest in CS; however, it could be assumed that one key factor in the success of implementing this learning area will be appropriate professional development (PD) that provides teachers with the confidence and experience to integrate CS effectively in classroom activities.

Australia is one nation leading a recent wave to introduce CS education into schools (Gander *et al.* 2013; The Royal Society 2012; Wilson and Guzdial 2010), with the release of the Australian F (Foundation, also known as Kindergarten) – Year 10 DT curriculum (ACARA 2012). The learning area explicitly introduces learning objectives relating to the cultural impacts of technology, computational thinking, the use of digital systems and data, as well as visual programming concepts and practice. While this is an exciting time for CS education, it also raises many big challenges. These challenges include the preparation of teachers to implement the learning area, teacher PD in domain knowledge, the development of resources and pedagogy for F-10 CS education, and more importantly, a need to address these challenges at a national scale. To address some of these issues, it has been recommended that the CS education community can assist by working with primary and secondary administrators (Barr and Stephenson 2011) to guide how to appropriately and effectively integrate new concepts into existing content and pedagogical knowledge and also how to integrate the learning area into classroom. As a university, how can we contribute to this national issue to prepare teachers to implement new computing curricula through scalable and cost effective solutions that could be adopted in other contexts?

MOOCs have been adopted as a means to deliver content (usually freely) across distributed environments to anyone with an Internet connection and computer. Further, social media have been integrated and offer the potential for supporting online community building, collaboration and knowledge sharing, despite learner locations. This approach to large-scale learning may provide a means to deliver content for free and allow teachers to develop a community where teachers can engage in flexible PD. Although online learning is not a new concept, little is known about what constitutes effective MOOC design, particularly for a purpose such as teacher PD.

In this paper, we describe an approach to develop a free teacher PD MOOC targeting the Australian DT learning area. This paper describes our MOOC design and findings about our participant audience and their engagement and experience in the course.

Literature review

New computing curricula

Over the past decade, the need to rethink our education systems in terms of the treatment of CS and information technology has gained global attention (Gander *et al.* 2013; Seehorn *et al.* 2011; The Royal Society 2012). Encouraging students to engage in current technologies and participate as creators of future technologies requires more than teaching the fundamentals of digital literacy (The Royal Society 2012).

There is a growing awareness among the CS education community that we must also teach computational thinking, the problem-solving processes and intellectual practices needed to understand the scientific practices that underpin technology. Without this, we face the risk of our youth being placed in the position of consumers of technology produced elsewhere, unable to actively participate as producers and leaders in this field (Gal-Ezer and Stephenson 2009; Gander *et al.* 2013). Recent reports from the United States and Europe have argued that it is essential that children be exposed to CS concepts and principles from the very start of their education so that every child may at least have the opportunity to learn computing at school (Gander *et al.* 2013; Wilson and Guzdial 2010).

New computing curricula introduced in England (Department for Education 2013), Australia (ACARA 2012), and New Zealand have been developed to introduce computing into schools; however, little has been done in these regions to prepare teachers to *implement* the new curricula. The challenges faced by the adoption of new curriculum are extensive and previous reforms have found that teachers feel left alone to struggle with the challenge of implementation (Park and Sung 2013), an issue likely to be more challenging when it is a completely new learning area. A consultation report, with feedback from industry, educators and the general public in Australia (ACARA 2013), has identified significant concerns relating to teacher PD, appropriate pedagogy and resource availability, particularly within the early years (Barr and Stephenson 2011). The report also identified concerns about the ability of teachers to readily make connections between DT and other learning areas. Similarly, rapid curriculum implementation in New Zealand and the United Kingdom have resulted in concerns for how prepared teachers were to implement the new learning area. If teachers are not prepared to teach computing, and lack the appropriate pedagogy and resources, how can we ensure that this global movement of CS into schools is effectively administered to have the desired impact on student learning and interest?

Discussions taking place in Europe (Gander *et al.* 2013) and the United States (Seehorn *et al.* 2011; Wilson and Guzdial 2010) suggest it may not be long before similar changes are introduced in other international contexts, bringing these issues to a global scale. The links between higher education, industry, educators and the schooling sector are identified as crucial to implementing new computing curriculum (Barr and Stephenson 2011); however, these connections are also required at a global scale, with various nations sharing strategies and approaches for preparing teachers to implement the new computing curricula. Although curricula may differ, the CS, computational thinking and programming concepts are still the same. Further, resources and lesson plans for teaching such content can always be adapted for particular learning objectives. Sharing approaches to PD in CS education, including design, content and pedagogy as well as findings from teachers' experiences and engagement will assist in moving implementation strategies forward.

Teacher PD

Despite significant changes in education with the emergence of technologies and software in recent decades, online teacher professional development (oTPD) has not necessarily adapted to such innovations (Brooks and Gibson 2012). In a review of teacher PD, teachers expressed that programmes were typically insufficient and too short to deal with many aspects of curriculum reform; resulting in authors advising that PD needs to be systematic, on-going and developmental rather than one-off

sessions (Park and Sung 2013). Currently, online modes of delivery and online communities may offer a means to maintain on-going development.

In a review of the literature, Brooks and Gibson (2012) describe PD models as falling along a continuum: beginning with face-to-face PD, to online teacher professional development (oTPD), which is structured Internet-based learning, and finally, technology-mediated professional learning (TMPL) which is a blend of both in a flexible nature. Although, there are more models of online learning to potentially work with (Harasim 2012), we have selected the approaches (oTPD and TMPL) proposed by Brooks and Gibson (2012) to explore. In Table 1, we compare the characteristics of oTPD and TMPL, with MOOC approaches, which will be discussed shortly.

However, which approach is suitable for delivering PD where participants must acquire new content knowledge, develop pedagogical knowledge and view exemplars linked to curriculum objectives? The approach of the TMPL model may work for promoting teacher professional learning networks and the co-creation of resources: an issue needing to be addressed in F-6 DT education. The model of TMPL acknowledges that teachers are not only drawn by the technological affordances of an online space but that they are participating in teacher communities or within online spaces and social network technologies out of specific curricular needs or collegial support. Essentially, TMPL also supports 'just-in-time' learning by seeking support in online professional networks, which can be an advantage for teachers needing support during implementation periods. However, for a new learning area some content will need to be delivered in logical sequences aligned with curriculum learning objectives, which will require the adoption of a more structured pace, but with a flexible and networking aspect to accommodate teacher workload and existing knowledge of the topics.

A key reason that teachers become engaged in online professional learning, across a number of studies reviewed, was due to the strong connections among the curriculum, student learning and teaching approaches (Brooks and Gibson 2012). How can we adopt a balance of these approaches to scale-up teacher PD nationally? One option is to look at massively open online courses (MOOCs) and approaches to delivering courses at such a large scale, to see if we can align the methods to design a course that meets teachers' needs of connecting the curriculum, student learning and teaching approaches.

Massively open online courses

MOOCs offer one means to deliver education at a broad scale to individuals with technological means and Internet access. Although online learning is not new, it has been argued that the difference between online learning and MOOC environments are the (1) combination of teaching approaches course instructors use, (2) massive levels of participation and (3) openness (Glance, Forsey, and Riley 2013). Previously, technology-driven education has seen many names applied to describe this mode of learning, such as distance education and e-learning (Rudestam and Schoenholtz-Read 2010), making existing research about online learning useful for informing potentially best practice in MOOCs. However, consideration towards the factors that differentiate MOOCs needs to be taken into account when selecting tools and approaches. Technologies and approaches selected need to cater for the 'openness' and 'scale' of the MOOC audience (Glance, Forsey, and Riley 2013).

Table 1. Characteristics of MOOC and online professional development approaches.

| | oTPD | xMOOCs | TMPL | cMOOCs |
|---------------------|---|--|--|--|
| Structure | Structured, Internet-based learning. | Structured, Internet-based learning, including video content, online quizzes, peer or self-assessment. | Less structure – ‘just-in-time learning’. | Less structure – distributed networks. |
| Focus | Online learning, self-directed study, course or module completion. Individualised models and cohort models. | Online learning, self-directed study, course completion. Some courses offer optional levels of engagement. | Blend of online and face-to-face learning. Social media (created and teacher-initiated groups). | Online-networked learning. Registrants create knowledge and engage in online publishing activities and self- or peer assessment, depending on personal learning goals. Distributed networks. A common platform or content aggregator may be used as a hub. |
| Platform | Learning management system (LMS) as core. | Core platform for content delivery and modules – perhaps through a service provider. | Online learning and face-to-face blended. Teacher-initiated networks and online communities. | Connectivist, communities of practice |
| Learning principles | Behavioural, mastery learning and knowledge-retrieval. | Behavioural, mastery learning and knowledge-retrieval. | Connectivist, constructivist. | Connectivist, communities of practice |
| Teacher emphasis | Content and delivery. Discussion forums for support. | Content and delivery. Discussion forums for support. | Encouraging networking (face-to-face and online), co-construction of knowledge and community support networks. | Encouraging online networking and professional community formation and co-construction of knowledge. |

Typically, two different types of MOOCs have been identified, one being based on existing university courses that embrace the use of videos to deliver content and online assessment ('xMOOCs') (Glance, Forsey, and Riley 2013) and the other courses based around online communities and connectivist principles called 'cMOOCs' (Siemens 2005, 2012). The characteristics of both formats are described in Table 1. In reference to the two PD approaches previously mentioned, xMOOCs are similar to oTPD and cMOOCs are similar to TMPL. However, there are also a number of 'hybrid' MOOC versions surfacing that combine a mixture of xMOOC and cMOOC approaches, blending a structured pace with a focus on participant-led communities, such as EDMOOC by Coursera, and MOOC-EDs introduced by the Friday Institute (Kleinman, Wolf, and Frye 2013).

Enrolment in MOOCs have reported significantly high enrolment rates, with edX and MITx reporting a total of 841,687 registrations from the fall of 2012 to the summer of 2013 across a number of their courses (Ho *et al.* 2014). In that year, 43,196 participants earned completion certificates. On average, there was a 5.17% completion rate across the courses, with a 9% completion rate for those who went beyond 'enrolment' in the course. A typical measure of completion within xMOOCs is the completion rate for those that complete half or more of the course, known as *explorers* – edX and MITx report a completion rate for explorers of 54%. A supporting component of xMOOCs are the community forums, which have seen engagement anywhere from 6.5 to 25.7% with an average of 7.9%.

In comparison, cMOOCs measure enrolment based on members who 'subscribe' to the course via mailing lists or by signing up to the course platform. cMOOC enrolment figures have been found to be ranging from the hundreds to the low-thousands. Researchers of cMOOCs typically report engagement through social media activity, for example, in MobiMOOC, the course generated 1827 discussion threads, 1123 Twitter contributions and 335 links shared on Delicious (de Waard *et al.* 2011). However, closer analysis reveals that a small core of participants generates activity. For example, in CCK11, 700 individuals enrolled for the course but researchers identified that only 18% ($n=126$) of registrants were actively involved in the course (Kop, Fournier, and Mak 2011) and in PLENK10, of the 1616 'subscribed' participants, only 40–60 individuals were identified as contributing actively. Similarly, in the course, First Steps in Teaching and Learning (FSTL12) over 200 individuals signed up for the course (Roberts 2012) but only 60 registrants actively participated throughout the 6 weeks and only 14 undertook assessment and received a certificate.

The literature so far suggests that the majority of registrants are professionals and enrol in the courses for PD, personal interest and to develop knowledge (Belanger 2013; Belanger and Thornton 2013; Breslow *et al.* 2013; Evans-Cowley 2013; Fini 2009; Miller and Odersky, n.d.). Such findings suggest MOOCs are a potentially appealing option for those interested in undertaking teacher PD; however, are they an effective approach for delivering teacher PD and what constitutes effective design? Taking a leaf from the MOOC approaches and what we know about oTPD and TMPD, we consider the development of an open online course in CS education, aligned with the Australian curriculum learning objectives so that teachers are able to easily put into-action lesson ideas and make connections between CS, *content and pedagogy*.

While research has begun to pedagogically evaluate MOOCs (Glance, Forsey, and Riley 2013), there is a great deal more required in terms of understanding the online

courses and how learners operate. Participants reportedly struggle with the overwhelming abundance of information and requirement of self-directed learning (Kop 2011; Mackness, Mak, and Williams 2010; Roberts 2012). This brings us to our research question:

How can we design a MOOC, as a large-scale professional development approach to address the challenge and support teachers with the implementation of a new computing curriculum?

In this paper, we describe our design process and rationale, followed by preliminary results relating to participation, engagement and teacher experiences. We conclude the paper with directions for future research and online PD in this field.

Context and design

The context

In Australia, 'primary school' includes the first year of school, called Foundation (F) followed by year 1, and so on, until year 6 or 7, (depending on the state) and secondary school (also known as high school) includes years 7 or 8 to year 12. In 2013, there were a total of 9,393 schools in Australia (Australian Bureau of Statistics [ABS], 2013). It is estimated, in 2011, that there were 123, 600 primary teachers in Australia, with around 80% engaged in classroom teaching.

The average age of primary teachers is 42.1 and 44.5 for high school teachers, with leadership roles being held by those around 50 years of age (Cordova, Eaton, and Taylor 2011). In Australia, the teacher workforce is predominately female, particularly in the primary years (81% of primary teachers and 57% of secondary teachers). In Australia, teachers are reportedly spending 46 hours per week on all school related activities and about 8 or 9 days a year towards professional learning (Cordova, Eaton, and Taylor 2011).

The Australian primary and secondary school system is undergoing a significant period of change, with the introduction of the Australian Curriculum. In 2014, the Australian Curriculum Assessment and Reporting Authority (ACARA) released a series of curricula for learning areas, with DT being one F-10 learning area, awaiting final minister endorsement. The new Australian curriculum has been developed recently to provide a national framework for states and territories, which can be implemented at their discretion.

In the DT curriculum, students learn CS knowledge and skills, and have opportunities to create digital solutions through designing and visual programming. However, a significant focus is on developing computational thinking, logic and problem-solving capabilities (ACARA 2013). The curriculum focuses on developing knowledge of digital systems, information management and the computational thinking required to create digital solutions. The core is the development of computational thinking skills: problem solving strategies and techniques that assist in the design and use of algorithms and models. The Australian Curriculum describes the nature of learners and curriculum across three broad year-groupings: Foundation to Year 2 (ages 5–7); Years 3 to 6 (ages 8–11); and Years 7 to 10 (ages 12–16). In the senior years (Year 11 and 12), students may select from specialised strands in the DT learning area.

Australian primary school teachers are typically generalist teachers, with 80% reportedly teaching in generalist classrooms (Cordova, Eaton, and Taylor 2011), trained to teach across the various learning areas prescribed by their state or territory. Some teachers may choose to specialise in a particular learning area, such as Physical Education; however this depends on the school. Some schools are fortunate enough to have an ICT teacher but this is not typically the case for all, with only 6% ($n = 7,500$) currently teaching computing (Cordova, Eaton, and Taylor 2011). In Australia, 17% of teachers report having had some post-secondary education in computing, with only 8% having been trained in the practice of computing (Cordova, Eaton, and Taylor 2011).

In identifying a need to resolve the challenges introduced with the introduction of new computing curriculum, the need to provide teachers with appropriate support to ensure DT education is implemented effectively, and not perceived as 'digital literacy', we initiated the idea of a PD MOOC to support Australian teachers. The team working on the project involved three individuals from the CS education research group at The University of Adelaide. Team members on the MOOC development team included two CS education lecturers with significant experience in introductory programming, problem-based learning and CS education pedagogy. The third team member was a research associate with experience in CS education research and qualifications in education and primary school teaching.

Design approach

The development of the MOOC involved content development and the more technical development of the course website through course builder. Although Google Course Builder is a free platform, technical development (in the form of coding) was required to initially construct the course using the web guides, as well as support and maintenance, which could be done by one of the CS lecturers.

Unlike previous MOOCs, where the content may have already been available from existing university courses or through aggregating web information, F-6 CS education lacks resources and content and so the majority of effort involved developing web content, 5-minute (average) concept videos and worked examples that linked to the Australian curriculum learning objectives. The development of content was achieved through brainstorming sessions between the team, drawing on their various expertise, experience and knowledge. We drew on and adapted existing lesson ideas from existing organisations and initiatives such as CS Unplugged and Code.org, and drew on lesson ideas and approaches from education texts in other learning areas, such as Mathematics, Science and Literacy. We worked closely with curriculum developers and curriculum elaborations to align content and lesson ideas to the Australian DT learning objectives. As previously mentioned, forethought is essential in online learning as lessons cannot easily be adapted (Beetham and Sharpe 2007). To develop a MOOC that aligned with teacher needs we collaborated with a range of experts in education to gather feedback on lesson ideas and content. Figure 1 presents the out MOOC development process. We began the development of the MOOC in November 2013, with a launch date in mid-March 2014.

There were initial outlay costs associated with acquiring equipment to produce the course videos, such as lighting, software (Camtasia), a green-screen, a computer, microphone and video camera. As we were also based in a tertiary institution, we were

| 1. Review CS & MOOCs | 2. Develop & Design | 3. Deploy & Review |
|--|--|---|
| Review empirical studies and F-12 CS Education resources | Collate activity ideas and develop content to align with the learning objectives | Deploy the course modules. |
| Review MOOC designs and literature. | Record videos and write web content. Send content and activity ideas to teachers for feedback. | Monitor and gather social media activity, Course Builder Analytics, YouTube data, survey and reflections. |
| Approach teachers, computing experts, industry, organisations and research groups (local and international) for feedback and support in development. | Refine and adapt based on feedback. Develop the course and align teacher MOOC tasks. | Respond to teacher feedback. In-course. |
| Apply for ethics approval to conduct MOOC research. | Create course platform in course builder (technical) and social media spaces. | Review and re-design for a 2nd version. |

Figure 1. The MOOC development process.

required to purchase resources, such as blocks, craft materials, robots, and tablet applications, that could be used to showcase lesson exemplars. We also purchased education textbooks about teaching other learning areas so that we could make cross-comparisons and develop teaching and learning approaches for this new F-6 learning area.

Review of existing empirical literature

To determine what empirical pedagogical literature was available for CS education, we began by undertaking a semi-systematic review of research papers about CS education, implemented for children between the ages of 5 and 18, using Simon's system (2007) to determine (1) the subject matter taught; (2) the age group studied; and (3) data collection methods. Our review identified F-6 as significantly lacking empirical research and resources and, where research was undertaken, it was to measure student interest in CS as a career rather than the explicit measurement of learning and teaching developments (Falkner, Vivian, and Falkner 2014). Our findings suggest these are the year levels that urgently require teacher support and the development of learning and teaching resources and so to begin to address this national challenge, our course focuses on these year levels, F-6, by delivering CS content, aligned with the learning objectives.

Learning from our review of online PD and MOOC pedagogy, we designed an approach that combined both oTPD/xMOOCs and TMPL/cMOOCs. In doing so, we are able to not only deliver discipline content knowledge in CS combined with learning objectives and classroom activities but also create an environment where teachers can share knowledge and collaborate, as well as seek and provide ad-hoc advice. Adapting these approaches and using a MOOC allows for teachers to engage in professional learning, where they are part of a larger community, and are undertaking a series of modules that are relevant to their practice and are on-going (rather than a one-off event).

Course structure & topics

One of the challenges we identified was that, as a new learning area, the content would be unfamiliar and potentially overwhelming for many teachers. Therefore, the course was designed around a series of topics that align with the Australian curriculum, delivered in a logical order, suitable for someone learning CS for the first time.

Our goal in the first unit was to provide an introduction that showcases the application and creation of DT to solve real problems, with an equal representation of males, females and youth as creators to dispel misperceptions that CS careers typically involve ‘programming’, are for ‘males’ or ‘nerds’. Further, we wanted to establish different terminology for DT (e.g. computing and CS) and distinguish between *using ICT*, *digital literacy* and *creating and solving problems with digital technologies* (the latter being the goal of the curriculum). In unit 2, the more familiar topics of patterns (creating and continuing sequences and recognition) and data representation (collecting and representing data in different ways, with and without technology) were introduced because of the potential links to what teachers are already doing in Mathematics and Science. In subsequent units, we moved towards the use and application of data by computers and digital data as well as the introduction of more abstract concepts, such as algorithms.

For each unit, we introduced the topic (e.g. ‘digital systems’) and the Australian learning objectives relating to the topic. Each unit was broken into sub-topics and for each sub-topic a concept video was created (or an existing suitable video used) in which the concept was explained and supported with analogies and real-world examples. For each unit, two full lesson exemplars about the sub-topics and a web resource with various lesson activity ideas were included. Links were made to the Australian curriculum ‘expected outcomes’ for children and assessment strategies were suggested. The goal was to provide core content knowledge and lesson idea inspiration so that teachers could feel comfortable creating and thinking of their own lessons to meet the learning objectives. The sequence of units for the DT course is outlined in Figure 2.

| | Unit 1: Introduction | Unit 2: Data (Patterns & Play) | Unit 3: Data (Representation) | Unit 4: Digital Systems | Unit 5: Information Systems | Unit 6: Algorithms & Programming | Unit 7: Visual Programming |
|-----------------|---------------------------------|---|--|--|---|---|--|
| CORE | All lessons | All lessons | Data Encoding and Different Representations Digital Data: Images, Sounds and Text Example Activity 1 | Hardware and Software Computer Networks | Services Social Protocols | All lessons | What are visual programming environments Algorithm Design in Visual Programming Example Activity 1 |
| NON-CORE | | | Counting in Binary Example Activity 2 Activity Ideas 1 & 2 | Input and Output Example Activities | Searching Encryption Example Activities | | How to use Scratch Programming using a visual Programming Environment Example Activity 2 Activity Ideas 1 & 2 |

Figure 2. Unit topics and core and non-core components of the course.

We introduced algorithms and visual programming last, with the deliberate goal to demonstrate that CS is more than just programming and that the learning area can include ‘plugged’ and ‘unplugged’ activities. Further, we wanted to explicitly demonstrate the importance of teaching concepts and computational thinking and to have teachers establish a sound understanding of the concepts, prior to visual programming.

Teachers were provided with a list of ‘core components’ and optional components (identified in Figure 2), with the ability to select their own learning path, depending on their previous experience and interests. Teachers were encouraged to complete as many units as they wished, however were prescribed a ‘minimum’ for completion, which involved fully completing the core components as well as the tasks for the three following units: ‘The Introduction’, ‘Data – Patterns and Play’ and ‘Algorithms and Programming’. In addition, teachers were asked to complete an additional unit of their choice, including both core and non-core components.

Technology

We have adopted a series of online course ‘spaces’ and technologies that support particular aspects of the course. In Table 2, we describe the two main course spaces, the supporting online spaces that provided participants with additional resources and finally, our additional course spaces along with their features and intended purpose.

Assessment

Two assessment tasks were included: one as a peer assessment task based on the creation of a teaching resource and the last as a lesson plan portfolio. For each unit, teachers were asked to post a task on the Google+ community page for the course. These tasks were designed to be informal and promote the exchange of tools, resources and lesson ideas. In all cases, teachers were provided with three options so that we could have a variation of content being shared. For example, ‘Find and share a useful data source that teachers could use as a resource for classroom activities. Please share with the community, along with your ideas on how this data source could be used in the classroom’. The final task involved a ‘lesson plan portfolio’. Teachers were asked to create two detailed lesson plans or briefly described sequence of activities about any of the topics covered in the course. Participants were to share this to the Google Community page and peer-review two others’ portfolios with a post ‘comment’.

In terms of PD accreditation, each state and territory differs; however, teachers are able to secure recognition for PD hours if they demonstrate committed hours towards activities. Therefore, we created an optional completion document, whereby participants could formally submit their unit tasks and associated content to receive a certificate, linked to the Australian professional standards for teachers (Australian Institute for Teaching and School Leadership [AITSL] 2014).

The course was self-paced, allowing for teacher flexibility and the completion of modules in any order, at any pace. Unlike previous MOOCs, our concern was not ‘enrolment’ and ‘completion’ statistics. We were satisfied if teachers were enrolling to access and use the resources and content without formal completing tasks. We initially released the first three units, weekly from the 22 March 2014, and then

Table 2. The course ‘spaces’ and technology: rationale and purpose.

| Technology | Features | Purpose/rationale |
|---|--|--|
| Main course ‘spaces’ | | |
| Google Course Builder Platform | Course modules; Forum; Announcements (also emailed); Information About the Course; Additional resources; Details about Q&A Hangout Sessions. | This is the key platform for content, course information and task specifications. The core platform has an announcements page and a forum space for course questions. |
| Google+ Community | The community page was organised into ‘topics’, where participants post their Task or assignment activity for each unit. An additional topic was created to share additional resources and events. | This space allowed participants to network, share ideas and activities and to collectively build an online series of resources corresponding to topic areas. It allowed for community feedback and ‘ad-hoc’ learning. |
| Supporting online space for additional resources | | |
| Pinterest | ‘Boards’ with pinned content sorted into topic areas that aligned with the Course Units. | Pinterest is popular among teachers for sharing and ‘bookmarking’ teaching ideas and resources. Even without an account participants can view content. Participants could locate additional resources, lesson ideas & tools for F-10 DT Education. Content shared by participants also pinned to relevant board. |
| YouTube | ‘Concept videos’ about F-6 DT content and lesson ideas created and embedded into Course Builder module lessons. | Uploading content videos to YouTube allowed for us to access viewer analytics and maintain these for public access – an outreach initiative to promote and support CS Education. |
| Google+ On Air Hangouts | Google On Air Hangouts were scheduled weekly for the first several weeks. They were automatically recorded and uploaded to YouTube for later-viewing. Q&A question poll was on for viewer questions. | These hangouts allowed for instructors to build a connection with participants watching, discuss common course questions, respond to questions posted and showcase guest speakers working in areas of F-12 CS Education and outreach. |
| Additional course spaces | | |
| Twitter | To share research and news about F-12 CS Education. | To maintain wide user-engagement and promote the course & CS Education. |
| Google+ Page | To share research and news about F-12 CS Education. | To maintain wide user-engagement and promote the course & CS Education. |

released all subsequent modules at once, allowing educators to work at their own pace, through the units, up and until the 30 June 2014.

Data collection and analysis

We were able to obtain data about participant demographics, participation and experiences through various avenues, such as through YouTube analytics, Course Builder learner analytics, Google+ Community page and voluntary surveys. During the course, participants were informed through course information that their data may be collected and analysed for research purposes and to improve future courses. Data collection and analysis were approved by the University of Adelaide Human Research Ethics Committee.

Course results

Overall, we had 1378 people enrol in the course, via the course website, and 473 participants continued to connect to the Google+ Community page. Although not core requirements of the course, engagement with Twitter and Pinterest was reasonable; however, the majority of the engagement occurred with the course and in the Google+ community space (Table 3).

Demographics

In unit 2, participants were provided with a voluntary anonymous survey, requesting their location as part of an exemplar activity to be presented back to participants. Although the participation rate is low (n = 174), we can see that the majority of participants appear to be from South Australia, Queensland, New South Wales and Victoria. Advertising and visits generally covered these areas, suggesting that for future courses, more targeted advertisements and connections need to be made to Western Australia, Northern Territory and Tasmania (Table 4).

Unsurprisingly, with the majority of teachers being female in Australia (81%) and the average age of teachers being between 40 and 50, the majority of the cohort were female and between the age bracket of 45–64 (see Figure 3).

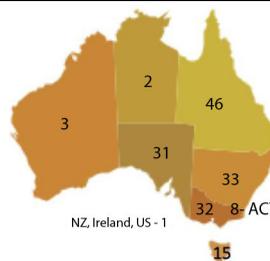
This is quite a different audience to the often ‘young professional male’ enrolling in edX and MITx courses. However, this demonstrates that we were able to target our intended audience and attract a female demographic that is significantly lacking in post-secondary courses and careers (Koppi *et al.* 2013).

Table 3. Enrolment/subscription statistics for the course ‘spaces’.

| Site | Enrolled/subscribed |
|---------------------------------|---------------------|
| Google Course Builder | 1378 |
| Google+ Community Page | 473 |
| Twitter | 294 |
| Pinterest (10 boards, 637 pins) | 336 |

Table 4. Number of participants by location (according to voluntary survey, n = 174).

| | |
|------------------------------|-----|
| Queensland | 46 |
| New South Wales | 33 |
| Victoria | 32 |
| South Australia | 31 |
| Tasmania | 15 |
| Australian Capital Territory | 8 |
| Western Australia | 3 |
| Northern Territory | 2 |
| International | 4 |
| Grand Total | 174 |



Participation and engagement

Of the 1378 enrolled in the course, 99 participants completed the course and 438 did not engage in the course any further than enrolling. As a result, we have a 7.2% completion rate, or 10.5% completion rate for those who went ahead and began the course. When considering completion rates, and measures of MOOC engagement, we consider engagement across all course components, and within core components specifically. Our completion rate overall was 7.2%, with a further 5.73% of participants exploring half or more of the course (without completion) and 56.39% of the participants completing less than half of the course. In terms of core components only, 8.13% of the participants explored half or more of the core components (without completion) and 52.3% of the cohort (group of MOOC participants) explored less than 50% of the core components. Our completion rate for explorers was 55.7%, and 46.9% when considering core components only.

Overall, across the course platform and the Google+ community, the completion rates were mostly in-line with what one would expect to see in MOOCs in terms of enrolment and completion. However, 34.3% of the cohort (n = 473) viewed and/or engaged with the online community – a significant increase in engagement over typical MOOCs. The completion rate relative to those that engaged with the community is 20.9%. A key motivating factor for this engagement was tying the course tasks in with the use of the Google+ community – a strategy that resulted in the co-creation of F-6 DT resources and lesson plans. Many of the teachers highly valued the resources being shared, the sense of community and the content that they

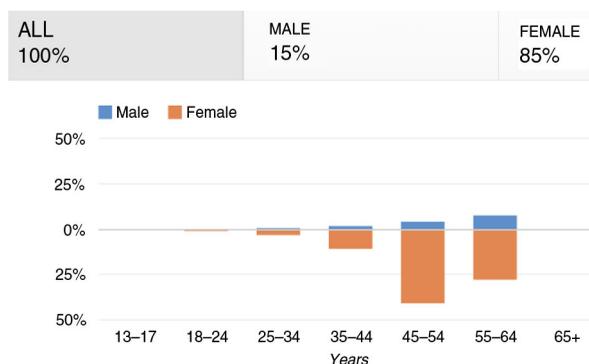


Figure 3. Gender and age bracket of cohort (via YouTube analytics – based on 4,565 views from 19 March to 29 June 2014, across 26 videos).

created together. One pre-service participant even expressed that ‘although I have finished this course, I keep coming back to make use of all the resources this community has to offer and to make further comments’.

In accordance with the participation and engagement described previously, we had a high number of viewers watching videos during the first unit ($n = 462$), slowly decreasing during each module (to 66 in unit 7). According to the YouTube analytics, the average video length created by the Computer Science Education Research group, was 5.8 minutes – ranging from around 1 minute to 11 minutes. This timeframe is typical of the ‘Khan Academy’ and xMOOC style video length and as the results suggest are the desired length as the average length watched was 4.37 minutes. The results suggest that small concept videos work; however, designers need to consider presenting important information at the beginning.

Experience

We invited course participants to respond to an online survey and to provide a reflection about their course experience after they had completed the final module of the course. We received 50 responses, which we acknowledge are biased towards participants who completed the final module. We were interested to know if the course had had any impact on their perception of CS and careers in CS, to which 26 of the 50 said that the course influenced their perceptions a great deal as they were not previously aware of career opportunities and 16 said they learnt more than what they had previously known.

The challenging topics reported by participants were around binary mathematics, with two teachers mentioning that the topic was challenging. One of the teachers said they felt they ‘still don’t really get it . . . [because] mathematical logic is not my forte’. While a number of participants reported the later topics being more challenging – from Unit 4 onwards – they thought they were ‘more interesting, provoking’ or ‘challenging but not difficult’. It appeared that demystifying some of the concepts and breaking down the language into examples and definitions assisted teachers in learning the content. One respondent, more comfortable in the topics, reported that: ‘Nearly all the modules were new in name but as I started to go through the MOOC I realised there was so much that I already did and understood’. For those who were new to the content, many had commented on the language and that the process of learning what concepts meant made implementing the learning area a lot less scary and ‘foreign’. For example, one respondent commented: ‘the language was the most challenging part [but once] you saw what ‘visual programming’ or ‘algorithms and programming’ encompassed it took the anxiety out of the topic’. Drawing on these comments suggest that making concept videos supported with everyday examples and exemplar activities can potentially assist teachers in understanding new discipline content.

Discussion

We have received generally positive comments from participants, via the questionnaire or informally via email. Teachers have expressed gratitude for the opportunity to participate in the course. One comment that stresses this appreciation was from a remote participant who expressed ‘Thank you for providing this course and also making it freely available. Living in country areas it is not always easy or affordable

to go to PD sessions. It has allowed me to network with other teachers who share the same interest and gather valuable resources'. However, in our current analysis we are lacking the perspective of teachers who have not completed the course and their experience. Understanding the motivations and experience of those who did not complete the last module may provide insight into how we can improve the MOOC and meet various needs and expectations.

The participation and engagement rates in this MOOC have been successful, in terms of the level of participation MOOCs generally receive. Particularly, the use of the Google+ community to share and build a repository of online resources, the short concept videos, the flexible learning pathways, the blend of content and exemplars as well as breaking down discipline language and concepts into relatable items, have proven to be positive design features of the course. A key to delivering content for a new learning area to educators appears to be finding the right balance of 'challenge' for teachers. New content and activities need to provide educators new to the discipline area with the fundamental knowledge but also opportunities for educators who are more comfortable with the learning area to extend themselves and act as mentors for others.

We have also learnt about areas for improving our course, namely, providing clear videos and instructions for participants who are not overly familiar with the technology and platform being used; reducing some of the compulsory workload for the courses or condensing the content in general. A concern expressed by a digital learning leader who was participating in the course was that they were lacking in confidence to develop and publicly share lesson plans in the community – something that they commented we often willingly assume teachers will do openly. This poses an interesting challenge: do we offer alternative tasks that are private, and if so, what trade-off would this have on the development of resources and community activity? In this case, we foresaw the need to develop F-6 resources urgently, with all participants benefiting from one another's resources.

While Google+ community appeared vibrant, we are unsure as to the extent that the community helped facilitate offline connections. Future work will investigate how to better facilitate these connections in subsequent courses through the use of technology and/or course design as well as the extent that the teachers continue to use the community for support.

Conclusion

We have trialled a MOOC approach to deliver PD to Australian teachers to support them with the implementation of a new computing curriculum. The process and course design described in this paper were intended to make a strong connection between the DT curriculum, CS content and teaching approaches (Brooks and Gibson 2012), as well as support ad-hoc interactions, resource development and flexible learning. We adopted a TMPL and hybrid-MOOC approach, with a core course platform that contained unit topics supported with content and videos, but also the integration of Google+ Community as a secondary course space. The community will remain open, providing an opportunity for continued use and for future cohorts moving through subsequent versions of this course to integrate with the existing community.

This MOOC paper should provide a framework and ideas for others encountering new curriculum implementation. The partnerships developed through this MOOC

have been valuable and the process of developing this course has cemented the importance of interdisciplinary and cross-institutional relationships to address the challenge of supporting teachers to implement new computing curriculum in terms of the development of professional learning courses, appropriate resources and for increasing awareness among the education community. International relationships are equally important as implementation of computing curriculum is a global challenge and sharing best practice and strategies will play a vital role in developing effective pedagogy, professional learning and resources so that the introduction of computing education can have the desired effect on student learning and engagement in CS career pathways.

Acknowledgements

We thank Google for supporting the development of this teacher professional development MOOC as well as the journal reviewers for their valuable feedback.

References

- Australian Bureau of Statistics (ABS) (2013). 'Schools', [online] Available at: <http://www.abs.gov.au/ausstats/abs@.nsf/Latestproducts/4221.0MainFeatures100042013?opendocument&tabname=Summary&prodno=4221.0&issue=2013&num=&view=>
- Australian Curriculum, Assessment and Reporting Authority (ACARA). (2012) 'The shape of the Australian curriculum: technologies', [online] Available at: http://www.acara.edu.au/verve/_resources/Shape_of_the_Australian_Curriculum_-_Technologies_-_August_2012.pdf
- Australian Curriculum, Assessment and Reporting Authority (ACARA). (2013) 'The Australian curriculum: technologies information sheet', [online] Available at: [http://consultation.australiancurriculum.edu.au/Static/docs/Technologies/InformationSheet – Technologies – February 2013.pdf](http://consultation.australiancurriculum.edu.au/Static/docs/Technologies/InformationSheet-Technologies-February2013.pdf)
- Australian Institute for Teaching and School Leadership (AITSL). (2014) 'Australian professional standards for teachers', [online] Available at: <http://www.aitsl.edu.au/australian-professional-standards-for-teachers/standards/list>
- Barr, V. & Stephenson, C. (2011) 'Bringing computational thinking to K-12: what is involved and what is the role of the computer science education community?', *ACM Inroads*, vol. 2, no. 1, pp. 48–54.
- Beetham, H. & Sharpe, R. (2007) *Rethinking Pedagogy for a Digital Age: Designing and Delivering e-Learning*, Routledge, New York.
- Belanger, Y. (2013) 'Saying goodbye to IntroAstro', *Duke Center for Instructional Technology*, [online] Available at: <http://cit.duke.edu/blog/2013/02/goodbye-to-introastro/>
- Belanger, Y. & Thornton, J. (2013) 'Bioelectricity: a quantitative approach Duke University's first MOOC', [online] Available at: <http://dukespace.lib.duke.edu/dspace/handle/10161/6216>
- Bell, T., *et al.*, (2011) 'Introducing students to computer science with programmes that don't emphasise programming', *Joint Conference on Innovation and Technology in Computer Science Education*, Darmstadt, Germany, pp. 391.
- Bock, S., *et al.*, (2013) 'Women and minorities in computer science majors: results on barriers from interviews and a survey', *Issues in Information Systems*, vol. 14, no. 1, pp. 143–152.
- Breslow, L., *et al.*, (2013) 'Studying learning in the worldwide classroom: research into edX's first MOOC', *Research & Practice in Assessment*, vol. 8, no. Summer, 13–25.
- Brooks, C. & Gibson, S. (2012) 'Professional learning in a digital age', *Canadian Journal of Learning and Technology*, vol. 38, no. 2, pp. 1–17.
- Cordova, J., Eaton, V. & Taylor, K. (2011) 'Experiences in computer science wonderland: a success story with Alice', *Journal of Computing Sciences in Colleges*, vol. 26, no. 5, pp. 16–22.
- de Waard, I., *et al.*, (2011) 'Exploring the MOOC format as a pedagogical approach for mLearning', *10th World Conference on Mobile and Contextual Learning*, Beijing, China, pp. 1–11.

- Department for Education. (2013) *The National Curriculum in England*, Department for Education Government of UK, Crown, Cheshire.
- Evans-Cowley, J. (2013) 'Update on Spring 2013 MOOC Enrollment, Participant Data for TechniCity', [online] Available at: <http://vt-uap.blogspot.com.au/2013/01/update-on-spring-2013-mooc-enrollment.html>
- Falkner, K., Vivian, R. & Falkner, N. (2014) 'The Australian digital technologies curriculum: challenge and opportunity', *Sixteenth Australasian Computing Education Conference*, Auckland, New Zealand, pp. 3–12.
- Fini, A. (2009) 'The technological dimension of a Massive Open Online Course: the case of the CCK08 course tools', *The International Review of Research in Open and Distance Learning*, vol. 10, no. 5, [online] Available at: <http://www.irrodl.org/index.php/irrodl/article/view/643/1402>
- Gal-Ezer, J. & Stephenson, C. (2009) 'The current state of computer science in US high schools: a report from two national surveys', *Journal for Computing Teachers*, [online] Available at: <https://csta.acm.org/Research/sub/Projects/ResearchFiles/StateofCSEDHighSchool.pdf>
- Gander, W., et al., (2013) 'Informatics education: Europe cannot afford to miss the boat', ACM, [online] Available at: <http://europe.acm.org/iereport/ie.html>
- Glance, D., Forsey, M. & Riley, M. (2013) 'The pedagogical foundations of massive open online courses', *First Monday*, vol. 18, no. 5–6.
- Harasim, L. (2012) *Learning Theory and Online Learning Technologies*, Routledge, New York.
- Ho, A., et al., (2014) HarvardX and MITx: The first year of open online courses, fall 2012–summer 2013, *HarvardX and MITx: The first year of open online courses (HarvardX and MITx Working Paper No. 1)*, Social Science Research Network: Social Science Electronic Publishing, [online] Available at: http://papers.ssrn.com/sol3/papers.cfm?abstract_id=2381263
- Kleinman, G., Wolf, M. & Frye, D. (2013) 'The digital learning transition MOOC for educators: exploring a scalable approach to professional development', [online] Available at: <http://www.mooc-ed.org/wp-content/uploads/2013/09/MOOC-Ed-1.pdf>
- Kop, R. (2011) 'The challenges to connectivist learning on open online networks: learning experiences during a massive open online course', *Special Issue – Connectivism: Design and Delivery of Social Networked Learning*, vol. 12, no. 3, pp. 19–38.
- Kop, R., Fournier, H. & Mak, J. (2011) 'A pedagogy of abundance or a pedagogy to support human beings? Participant support on Massive Open Online Courses', *International Review of Research in Open and Distance Learning*, vol. 12, no. 7, pp. 74–93.
- Koppi, T., et al., (2013) 'Addressing ICT curriculum recommendations from surveys of academics, workplace graduates and employers', [online] Available at: <http://www.arneia.edu.au/project/37>
- Lambert, L. & Guiffre, H. (2009) 'Computer science outreach in an elementary school', *Journal of Computer Science in Colleges*, vol. 24, no. 3, pp. 118–124.
- Liu, J., et al., (2011) 'A survey on computer science K-12 outreach: teacher training programs', *Frontiers in Education Conference*, Rapid City, San Diego.
- Mackness, J., Mak, S. & Williams, R. (2010) 'The ideals and reality of participating in a MOOC', *7th International Conference on Networked Learning*, Aalborg, Denmark.
- Miller, H. & Odersky, M. (n.d.) 'Functional programming principles in Scala: impressions and statistics', [online] Available at: <http://docs.scala-lang.org/news/functional-programming-principles-in-scala-impressions-and-statistics.html>
- Mykietiak, C., et al., (2012) 'cs4fn: a flexible model for computer science outreach', *Innovation and Technology in Computer Science Education*, pp. 297–302, ACM, Haifa, Israel.
- Park, M. & Sung, Y.-K. (2013) 'Teachers' perceptions of the recent curriculum reforms and their implementation: what can we learn from the case of Korean elementary teachers?' *Asia Pacific Journal of Education*, vol. 33, no. 1, pp. 15–33.
- Roberts, G. (2012) 'OpenLine project: final report', Oxford Brookes University, [online] Available at: http://openbrookes.net/firststeps12/files/2012/08/OpenLinefinalreport2012-08-10_merged.pdf
- Rudestam, K. & Schoenholtz-Read, J. (2010) 'The flourishing of adult online education: an overview', in *Handbook of Online Learning*, 2nd edn, eds. K. Rudestam & J. Schoenholtz-Read, SAGE, Thousand Oaks, CA, pp. 1–28.
- Seehorn, D., et al., (2011) *CSTA K-12 Computer Science Standards*, Computer Science Teachers Association, Association for Computing Machinery, New York.

- Siemens, G. (2005) 'Connectivism: a learning theory for the digital age', *International Journal of Instructional Technology and Distance Learning*, vol. 2, no. 1, pp. 1–9.
- Siemens, G. (2012) 'MOOCs are really a platform', *ELeanspace*, [online] Available at: <http://www.elearnspace.org/blog/2012/07/25/moocs-are-really-a-platform/>
- Simon, (2007) 'A classification of recent Australasian computing education publications', *Computer Science Education*, vol. 17, no. 3, pp. 155–169.
- The Royal Society. (2012) 'Shut down or restart? The way forward for computing in UK schools', [online] Available at: https://royalsociety.org/~media/Royal_Society_Content/education/policy/computing-in-schools/2012-01-12-Computing-in-Schools.pdf
- Wilson, C. & Guzdial, M. (2010) 'How to make progress in computing education', *Communications of the ACM*, vol. 53, no. 5, pp. 35–37.
- Zweben, S. (2012). Computing Degree and Enrollment Trends. Computing Research Association, Washington, DC, pp. 1–17.