flanz
Journal of Open, Flexible, and Distance Learning

Computational Thinking in Junior Classrooms in New Zealand

Jeanette Irons, Massey University
Maggie Hartnett, Massey University

Abstract

From 2020, the New Zealand technology curriculum includes computational thinking. The new curriculum content is being introduced to students from 5 years' old. In preparation for its introduction, online resources have been developed for teachers (including junior teachers who teach new entrants, up to Year 3), that contain progress outcomes, lesson plans, exemplars, and assessments. However, it's not clear whether New Zealand junior teachers are sufficiently prepared to teach computational thinking and what factors influence their preparedness to teach the new curriculum. This research explored the experience of a small group of junior school teachers in the year before the technology curriculum was officially introduced. Research findings highlight that factors (including professional development, assessment, schoolwide support, and time availability) influence the uptake of the computational thinking curriculum by teachers in New Zealand junior classrooms.

Keywords: computational thinking; technology curriculum; professional development; New Zealand; junior teachers; K–12

Introduction

From 2020, computational thinking has become part of the New Zealand school curriculum for students from 5 years' old (Ministry of Education, 2017). This is not a revised curriculum—it's a completely new addition to the technology curriculum. The purpose of the computational-thinking curriculum is to teach New Zealand students to be more than consumers of technology, and to develop a greater appreciation of existing digital devices and how to create innovative digital technologies (Bell & Roberts, 2016; Wing, 2006). The new digital-technologies curriculum content involves, among other things, teaching children how to sequence their ideas using algorithmic thinking, to use debugging strategies to solve problems, and to understand binary language and heuristics. The learning progressions envisioned in the new curriculum have been developed as a series of progress outcomes for New Zealand teachers to follow (Ministry of Education, n.d.-b) and introductory training has been made available online (see, for example Kia Takatū ā- Matihiko, n.d.; The Mind Lab, n.d.).

This article explores the experience of several junior teachers embarking on teaching the new computational-thinking curriculum content and the factors that influence their teaching.

Literature review

A definition of computational thinking

Computational thinking has been defined as "the thought processes involved in formulating a problem and expressing its solution(s) in such a way that a computer—human or machine—can effectively carry out" (Wing, 2017, p. 8). Systematic and logical problem-solving, creativity, evaluation, and understanding of the world we live in are all encouraged by computational thinking (Caeli & Yadav, 2020).

Computational thinking can foster higher-order thinking (Falloon, 2019). Computational-thinking education encourages active learning rather than passively using digital devices, and children as young as 5 have been observed problem-solving, thinking critically, collaborating, and exhibiting cognitive perseverance while undertaking coding tasks (Falloon, 2016). Many computational-thinking software programs have powerful potential (Grover & Pea, 2013), but until children understand how to manipulate these tools effectively they tend to engage only with easy and entertaining activities (Bell et al., 2009).

Drivers for introducing computational thinking to school curricula

Advocates have argued for including computational thinking in the curriculum since the 1960s (Caeli & Yadav, 2020) with little success. However, society is changing rapidly as a result of the increasing ubiquity of digital technologies in all facets of life, and the need to include computational thinking in educational curricula has become urgent (Grover & Pea, 2013). Computing jobs are currently amongst the fastest growing worldwide (Vlahu-Gjorgievska et al., 2018). Many existing jobs that were once considered "unskilled"—such as routine factory work, pattern-cutting, and electronics construction (Caeli & Yadav, 2020)—are now done by robotic machines that require digitally capable employees to set them up, repair them, and oversee their operation. These jobs require people with computational thinking skills and expertise (Grover & Pea, 2013).

Many countries, such as Australia, the United Kingdom, the United States, and New Zealand have introduced computer education, coding, computer science, dataology, and computational thinking into their technology curricula with the intention of increasing the availability of computer-literate employees (Caeli & Yadav, 2020; Grover & Pea, 2013). New Zealand is at the forefront of introducing computational thinking into new entrant (Year 0) classrooms. Existing investigations from K–12 (primary to secondary) classrooms tend to focus on students beyond 8 years' old (e.g., Burnett, 2006) or K–12 results are aggregated—resulting in the inability to specifically identify findings that relate to junior classrooms (e.g., Nouri et al., 2020). New Zealand Ministry of Education documents on this subject also target older children (see, for example Bolstad et al., 2012). Although computational-thinking pilot schemes have started in some New Zealand classrooms (Bell & Bell, 2018; Bell, 2018) and in some Danish schools (Vlahu-Gjorgievska et al., 2018), computational thinking education in junior classrooms across New Zealand is in its infancy.

Teacher preparedness

No matter how transformational a new curriculum or programme might be, studies worldwide have shown change will not occur if teachers are not committed (Hattie, 2017; Rogers, 2010; Yin et al., 2011). Resistance to implementing new aspects of a curriculum can be influenced by personal cost (Waugh, 2000), perceptions of benefit (Cviko et al., 2012), negativity or positivity towards the innovation (Ertmer & Ottenbreit-Leftwich, 2010), resources allocated to the new curriculum (Waugh, 2000), teacher support, and alignment with assessment (Gibson & Brooks, 2012).

Introducing technological advancements into a classroom usually involves upgrading hardware and software—and these requirements can affect implementation (Waugh, 2000; Yin et al., 2011). Apart from monetary costs, the time required to implement a new curriculum (Terrazas-Arellanes et al., 2016), time to learn new skills or software (Gibson & Brooks, 2012), time to search for and test resources (Sotiriou et al., 2016), and time to update and prepare planning documents (Blundell et al., 2016; Giffen Cheng et al., 2014) are important considerations for teachers. Waugh (2000) also noted the importance of school management proactively addressing the concerns of teachers regarding the personal cost of change.

School curriculum change is frequently initiated by administrators (Moltó et al., 2009). However, teachers are not passive absorbers and transmitters of teaching programmes, and their input about implementation is vital (Hunter & Keown, 2001).

Waugh (2000) highlighted the positive influence of teachers being part of the decision-making process in schools, while Moltó et al. (2009) emphasised the mismatch between governmental policy and actual practice when teachers were not involved. Successful leadership is a critical precursor to change (Agarwal et al., 1998; Baylor & Ritchie, 2002). Although change can be slow, having leadership actively involved with professional development and classroom teaching can influence the desired outcomes (Gibson & Brooks, 2012). Teachers need to know where to go to access support, and need to take an active role in introducing a new curriculum (Moltó et al., 2009; Waugh, 2000).

When teachers are prepared to introduce a new curriculum into their current classroom practice it is more likely to happen (Giffen Cheng et al., 2014). Moltó et al. (2009) and Waugh (2000) showed that a complete reorganisation of existing practices is necessary to ensure successful integration of new curricula. For example, if teachers can't easily find supporting publications, software, planning ideas, or curriculum support they are unlikely to change their existing teaching practices. Therefore assessment, timetables, and resources for learning (to name a few) are important considerations because they are influential in determining whether changes are enacted at the classroom level (Blundell et al., 2016; Giffen Cheng et al., 2014). With the recent introduction of the computational-thinking curriculum content, there's a gap in the research about the experience of teachers of 5–8-year-olds, and who are new to teaching computational thinking. This project investigates the experience of junior teachers (new entrant to Year 3) in New Zealand and therefore goes some way towards addressing the gap in the literature.

Research method

This research study is interested in the perceptions and experiences of junior teachers in relation to teaching computational thinking in New Zealand. Therefore, a qualitative research approach was undertaken to answer the research question: What are the factors that influence junior teachers' preparation for teaching computational thinking?

The research followed guidelines from the code of ethical conduct for research, teaching, and evaluation involving human participants (Massey University, 2018). A low-risk ethics approval was sought and granted before the research commenced.

Semi-structured interviews were used to collect the thoughts and experiences of junior teachers about computational thinking. This was the preferred data collection method because it allowed questions to be prepared in advance, and enabled the interviewer to pursue responses in more depth (Creswell & Plano Clark, 2018).

Each interview was audio recorded, transcribed using NVivo transcription software, and checked for accuracy. Initial themes began to emerge. A thematic analysis was undertaken with NVivo12, and commonalities between themes were analysed with NVivo query coding comparisons.

Context and participants

Seven junior teachers who were teaching children from new entrant to Year 3 (5–8 years' old) were interviewed. These teachers taught at three schools in central New Zealand. Pseudonyms have been used throughout this article to ensure confidentiality (see Table 1). Participants' quotes have been edited to follow written English conventions without changing their intended meaning.

Table 1 Participants' information

Teacher's pseudonym	Year level	Age of children
Sharon	Year 0 to 3	5-8
Sandra	Year 0 to 3	5-8
Jane	Year 1-2	5-7
Ruby	Year 0	5
Rian	Year 0	5
Mere	Year 0-1	5
Toni	Year 0	5

Results

The results section begins with participants' understanding of computational thinking. Next, the participant's attitudes to computational thinking are explored, and categorised into positive and negative statements as well as perceived benefits and challenges. This is followed by emerging themes about assessment, professional development, support, resources, and time.

Teachers' understanding of computational thinking

The participant's depictions of computational thinking were varied. Although all the teachers had heard of computational thinking, and most were teaching elements of it in their classroom, some found it challenging to put computational thinking into words and many were initially reluctant to do so. Those who had undertaken professional development had the most confidence in their understanding of the new curriculum, and were explicitly teaching it in their classes. Rian explained computational thinking as "how information is stored on the computer. So, computational things, and how we're programming, creating apps et cetera in the junior end". Rian also included detailed descriptions of lesson plans she had used with her new entrant class from "Tahi Rua Toru Tech", using number flash cards and mathematical concepts (see Tahi Rua Toru Tech, 2018).

Most participants expressed an opinion that junior children needed to use technology to learn about computational thinking; for example, Ruby said it is "getting them familiar with the tools they can use". Use of digital devices was the area in which there was resistance to including computational thinking in the classroom. Jane's comment exemplifies these concerns: "some kids had been too much on [digital devices] and they became quite bizarre, they won't get off and we had fights and we had behavioural problems". Mere's statement: "I don't think it's something I'd do with my kids because of the word computational", revealed her emergent understanding of computational thinking. Jane and Mere recognised that they knew very little about computational thinking, but both also acknowledged their belief in the importance of learning new technologies alongside the children.

Teachers' attitudes to computational thinking

Only one of the participants in this study was actively using downloaded lesson plans to teach computational thinking. Rian was integrating computational thinking with mathematics. Most of the other teachers had taught elements of computational thinking in the form of sequencing or exploring the Scratch app, but were not using computational-thinking language and concepts such as, "algorithm", or "debugging". Some participants were not aware they were teaching facets of computational thinking and only through the interview discussion did they became aware of the potential for developing current practices into effective computational-thinking lessons.

When exploring junior teachers' attitudes to computational thinking, more positive (56% or 10/18) than negative (44% or 8/18) statements were coded. Participants expressed many positive statements about computational thinking being necessary for the future. When Jane thought about the imminent technology curriculum change, she said, "Oh great, I think it's sort of second nature with most people now. Because we're immersed among it—if we don't learn it, we'll be left for dead"; while Sandra said, "I think there's potentially a lot of benefits". Toni felt that if she was asked to start teaching computational thinking tomorrow, she might scramble but would do it. However, negative statements were also recorded. Ruby said, "probably not at my particular age because for a lot of them [students] it's actually just being able to use the programs and things like that", and Mere said, "for me personally, I want my kids to read and write and count and all that stuff, I suppose that's always at the forefront". Table 2 categorises each participant's positive reasons for including the computational-thinking curriculum in the classroom; for example, it incorporates the key competencies and enables exciting lessons. Negative aspects of teaching computational thinking were expressed as it not being appropriate for the age, and using devices for extended periods of time.

Table 2 Positive and negative statements about computational thinking

Positive statements	Negative statements
Necessary addition to curriculum	Time taken away from core curriculum
Exciting lessons	Not appropriate for age group
Impressive learning	Using devices for extended periods of time
Incorporates core curriculum	Lack of teacher knowledge
Incorporates key competencies	Pressure to concentrate on core subjects
Potential	Too much play (from parents' perspective)
Natural progression	
School priority	

Perceived benefits and challenges to computational thinking

Participants believed computational thinking was a necessary addition to the curriculum and that it could, for example, be incorporated in core subjects and be used to teach the key competencies. They felt computational thinking reflected the interests of the children and the way the world was heading (e.g., the collaborative but distributed nature of work). The following comments by Jane and Rian reflect those made by the group as a whole:

> I understand the need for coding, it works particularly well when you've got tuakana teina [older and younger children working together] (Jane)

> students loved it . . . [we can] incorporate mathematics . . . [it can be part of] literacy or language. (Rian)

Table 3 is a complete list of benefits identified by participants.

Table 3 The benefits of teaching computational thinking

Benefits	Example
There is a need	"anything to do with thinking and problem solving or around key competencies, I think is always going to have a place and be incredibly worthwhile" (Sharon)
Children can lead because of prior knowledge	"The big kids, because they're wonderful teachers, and they know it way better than I do" (Jane)
They are already immersed in it	"if we don't learn it we'll be left for dead" (Jane)
Low cost	"because again it's kind of free" (Sharon)
Enjoyable learning	"students loved it" (Rian) "bringing in something fresh" (Sharon)
Can be independent	"Once they're familiar with it, it might be something that they can access on their own in play time" (Sharon)
Incorporates maths, literacy, and language	"That's something definitely to bring into everyday literacy" (Ruby)
Creativity	"and you've got the creativity" (Rian)
Incorporates key competencies	"you've got the key competencies" (Rian)
Reflects societal changes	"where the whole world in general is heading" (Ruby)
Links to children's curiosity	"How we can help those children with their interests" (Ruby)
Incorporates different types of learning	"But it's definitely even back to the basics of spatial awareness in the understanding of body movement and things like that" (Sandra)

Challenges more broadly associated with digital technologies were often expressed as challenges to introducing computational thinking in the classroom—such as lack of devices, learning to use programs, and lack of funding. Some participants' concerns arose from a lack of understanding of how computational thinking could be taught, and how to incorporate it in an already busy timetable. Others said they thought computational thinking was more appropriate for older children, or their school focused more on technology use by older children. Several participants expressed doubts about their relative inexperience in a school or teaching a particular age level, and felt they had to concentrate on the basics (i.e., literacy and numeracy) before they introduced new content. (See Table 4 for a complete list.) Rian was the only participant who could not think of any challenges associated with teaching computational thinking to juniors. Mere and Toni spoke only about the challenges of teaching computational thinking, and the other participants identified both benefits and challenges.

Table 4 The challenges of teaching computational thinking

Challenges	Example
Lack of equipment	"Probably the equipment side" (Toni)
Getting kids beyond just playing	"we risk that the kids play on it" (Jane)
Time spent on devices	"and you'll hear 'oh I don't think my kids should be on iPads for this long' " (Mere)
Lack of funding	"if we had a fountain or mountain of money" (Sharon)
Priorities	"as always it's a bit of fitting it in and prioritising" (Sharon) "ensuring you're doing it, rather than letting it fall off" (Sandra)
Lack of teacher knowledge	"I was terrified when they said I had to do the Scratch Junior" (Jane)
Learning to use programs etc	"It'd be great if someone can come in with the technology and show me how it's done" (Mere)
Linking to age group	"the apps and things have to be really appropriate" (Jane)
Linking to children's prior knowledge	"But it's how? How we do it to actually fit with what they know and what they can do at the different ages and stages?" (Ruby)
Taking focus away from core curriculum	"I want my kids to read and write and count and all that stuff. I suppose it's always at the forefront" (Mere)
Time for lessons and planning	"it takes up too much of part of the day" (Jane)
Making sure it has a purpose / authentic integration	"I think it has to have a real purpose" (Jane)
Language (learning it and teaching it)	"Probably language" (Sandra)
New staffing challenges	"Plus we've had a few changes. We've got two new teachers that have started this term. And then I'm new into the juniors anyway and I stepped in as team leader" (Sharon)
Making sure it doesn't get missed	"it's finding ways that I can authentically fit these sorts of things within the numeracy and literacy" (Sharon)
Focus on older children / complexity of curriculum	"I wouldn't be jumping straight in with the younger ones" (Sharon)

Assessment

All participants struggled to identify an existing assessment that could be used effectively when teaching computational thinking. All participants indicated there were no resources currently available in their classrooms to assess computational thinking, although they felt existing tools (such as Seesaw—a digital portfolio) could be used for capturing evidence of participation in computational-thinking activities. Jane could imagine creating a database to record children's progress in computational thinking, and Rian imagined competency in using technology could be measured. But Sharon pointed out the difficulty of measuring thinking: "that's quite a challenging one, because sometimes you'll see higher order thinking in certain areas and not in others". Sharon also expressed that, without an effective assessment programme, computational thinking could be missed out of the curriculum: "we're very busy so I know that my priorities are often down to deadlines and accountabilities and things".

Professional development and support

None of the participants had attended any recent school-led professional development about computational thinking. Rian had found information online, downloaded classroom lessons and resources, and enrolled for an online digital-readiness programme, while Sharon said she had been "just having the odd look online". Table 5 shows that six of the seven participants, namely Rian, Sandra, Sharon, Toni, Jane, and Ruby felt well supported by expert technology teachers in their schools, although Ruby said she would probably search for an answer online before asking the expert teachers. Mere was the lead teacher in ICT (at her level) at her school, and said she would also search online if she needed additional support.

Table 5 Professional development experience of teachers

Name	PD	Support available
Mere	Self-directed online	Online
Rian	none	Lead teacher in school and online
Ruby	none	Online or lead teachers at school
Sandra	none	Lead teacher in school and online
Sharon	none	Lead teacher in school and online
Toni	none	Lead teacher in school
Jane	none	Lead teacher in school and online

All participants felt supported by their school's leadership team, but although allwere positive about attending school-organised professional development (if available), they didn't all think computational thinking was sufficiently important to prioritise learning in their own time. The strong culture of reliance on experts in schools was a recurring theme. Sandra and Sharon both admitted that, although their lead technology teachers had emailed them computational-thinking information and links, neither had accessed them because of lack of time, priority, or motivation. Sandra said, "I haven't been proactive in making the most of what's available". Toni (who admitted knowing very little about computational thinking) had been gathering information from other teachers in her school who were currently undertaking computational-thinking professional development. She said: "people are just talking about these things and it's starting to become the conversation". Rian's self-directed learning and her clearly articulated understanding of computational thinking were not acknowledged (perhaps not known) by other teachers in her school. Rian was not cited as the teacher expert in computational thinking and she spoke of going to the technology lead teacher if she needed assistance.

Resources

Most participants had ideas about the resources they felt were needed to teach computational thinking. Jane knew about a set of coding books that she had seen used at a previous school, and she thought they would be helpful. Mere knew of the Bee Bots (small robotic bees that can be programmed to move forwards, backwards, left, and right), but was limited by her own understanding of how to use them beyond playing with them. Mere knew of a lot of resources that she thought were suitable for computational thinking but targeted older children, such as 3D printers and Makey Makey (a digital invention kit). Rian was also aware of tools for older children but she was using existing resources, such as grids and rockets painted on the concrete, to adapt ideas she had discovered online. Rian felt she didn't need any additional resources, but she might need more tools in the future. Ruby and Toni knew of no existing resources (other than iPads) for teaching computational thinking. Ruby and Toni felt they would need to know more about computational thinking before they could identify the resources they needed. Sandra had heard of a lot of resources for intermediate-aged children (Year 7–8, or 10–12 years' old) and wanted to explore Strawbees (a construction kit with digital programming potential), which she had seen advertised for junior classrooms. Sharon was preparing to introduce Scratch Junior to her Year 3 students but didn't think it was a suitable tool for the 5-year-olds. She had a lot of ideas for additional resources to teach computational thinking (such as circuitry, magnets, and Strawbees), but acknowledged that they did not have "a fountain or mountain of money". Sharon also knew of many resources that targeted older children, and a lot of resources that were fun (such as "virtual reality goggles and the spheres that roll around and go") but she questioned their educational value. Sharon said the thinking could be taught "without necessarily needing new technologies".

Time

The issue of time was highlighted by many of the participants. Teachers were concerned that computational thinking would lead to students spending too much time on devices. Two participants mentioned that their school had already removed technology from their "play to learn" sessions because the children chose to use digital devices rather than engaging with peers. Another participant said she limited the time that children had with computer technology to 10–15 minutes. Time taken away from core curriculum subjects was another repeated concern, as was how to link computational thinking with literacy and numeracy so that it wasn't another thing to have to add to the timetable. Several participants felt that the time required for children to learn and the time required to implement computational thinking would need to be substantial. The word "priorities" was used repeatedly. Sharon said, "there's always more I want to do, but can't fit in". The teachers also lamented their personal lack of time to learn. Each teacher was in a unique situation but they all had full schedules, including other professional development, leadership roles, and personal commitments. The time needed for planning was also mentioned as a future goal. Rian, the one teacher who had taken time to personally develop her skill, had also been out of the classroom for two terms in a management support role.

Discussion

Based on the findings presented in the previous section, two overarching themes emerge; namely, teacher knowledge and resources.

Teacher knowledge

Although computational thinking was not officially part of the curriculum until 2020, government expectations in 2019 were that teachers would engage in professional development in anticipation of its introduction. The results of this research show that although they were willing to teach computational thinking, most of the junior teachers in this sample felt they were not equipped to do so. Many of them believed computational thinking required a digital device, although this is not stated in the computational-thinking curriculum (Ministry of Education, 2017).

Teachers' understanding of computational thinking, the benefits they perceive of teaching it, and the challenges they face when teaching the new content are important. The research will contribute to developing an overall understanding of junior teachers' experience. Findings indicated the teachers are aware of the new curriculum, which is an important starting point for future conversations about computational thinking in the junior classroom.

Five of seven participants in this study perceived benefits to teaching computational thinking, but six of them also identified challenges. Many of the negative statements indicated a lack of understanding of what teaching computational thinking entails for junior students. The common concern about how much time children spend on devices assumes that computational thinking needs to be taught with digital technologies. However, in the New Zealand curriculum, Computational Thinking Progress Outcome One (which is the most relevant for junior teachers) specifically states the use of non-computerised tasks: "In authentic contexts and taking account of end-users, students use their decomposition skills to break down simple non-computerised tasks into precise, unambiguous, step-by-step instructions (algorithmic thinking)" (Ministry of Education, n.d.-a, p. PO1). Furthermore, there is a perception that access to devices should be limited because they result in less peer engagement, but obsessive behaviour may be overcome when children can collaborate on computational thinking projects (as noted by Falloon 2016, 2019).

In this study, teachers' attitudes to computational thinking appear to influence their preparation for teaching computational thinking. Their lack of knowledge and understanding needs to be addressed to reduce the challenges perceived by junior teachers when teaching the new curriculum content. In addressing teachers' lack of knowledge, it can reduce the resistance to its implementation—as others have noted when new technologies are introduced (Waugh, 2000; Yin et al., 2011).

The results of this research highlight that New Zealand junior teachers need to participate in ongoing professional development if they are to introduce computational thinking in classrooms successfully. The new curriculum is not a revision of the existing technology curriculum, but an addition; therefore, it is reasonable to expect junior teachers to require specialised training to teach it effectively. Although computational-thinking training and resources are available online, the participants in this study had not been encouraged or motivated, nor were they given, or had, time for self-directed professional learning opportunities. They talked about relying on the "experts" in their schools rather than taking advantage of online resources. As a result, insufficient subject knowledge appears to influence their preparation for teaching computational thinking. Two participants spoke of learning new technologies alongside their students, but to extend learners, teachers need to both increase their own computational thinking knowledge and become experienced users of the technology. Knowing the curriculum is a necessary prerequisite to teaching it (Kereluik et al., 2013; Moats, 2010; Tell et al., 2000).

For change to occur, teachers need to feel supported (Ertmer & Ottenbreit-Leftwich, 2013). This research indicated that the support these participants experienced did not necessarily transfer to new teaching practices (as shown by Agarwal et al., 1998). All of the participants felt they were well supported and they knew where access information, but unless they had actively and independently pursued these resources, they were not teaching computational thinking.

This study has suggested a possible disconnect between government policy, leadership, planning, and teachers' practice for the participants involved. Ruby and Toni openly expressed their lack of computational-thinking knowledge and contribution to introducing the new curriculum. Lack of involvement in school planning and implementation may be a factor influencing the lack of preparedness for teaching computational thinking apparent in this study. Waugh (2000) and Moltó et al. (2009) showed the importance of teachers being involved at the decision-making level when a new curriculum is introduced.

Resources

Lack of specific resources—including assessment, specific digital tools, and time—were also highlighted by participants as barriers to teaching the new curriculum content.

The most salient theme identified by participants was the perceived lack of alignment between the computational-thinking curriculum and existing assessment practices. For change to be implemented, a new curriculum needs to align with current assessment practices or include appropriate assessment (Waugh, 2000). The perceived lack of readily available assessment practices and examples may affect the success or otherwise of the introduction of the new curriculum content in junior classrooms. The New Zealand Ministry of Education has developed a document called "Indicators of Progression" (Ministry of Education, n.d.-b), which includes assessment for computational thinking with achievement objectives. The participants' lack of knowledge about this document (which is available online) suggests the need for further publicity as part of professional development for computational thinking.

One challenge shared by all participants was the pressure of time, which led busy teachers to decide which things to prioritise. Rian had dedicated professional development time out of the classroom. In line with other research (such as Moltó et al., 2009) these results suggest that, for computational thinking to be prioritised by teachers, schools will need to support professional development that includes dedicated time out of the classroom. Schoolwide timetabling also needs to reflect the importance of computational thinking if these teachers are going to consider it a priority (Waugh, 2000).

The teachers' knowledge of digital resources influenced their preparation for teaching computational thinking in this study. Two participants were unaware of any digital resources that aligned with the new curriculum, while several teachers knew of resources but didn't feel they were appropriate for junior classrooms. Two teachers spoke of resources (such as Strawbees) that were being marketed as appropriate tools for teaching computational thinking in the junior classroom, but the participants questioned the expense.

The perceived lack of resources available to teach computational thinking may be an influential factor but until these teachers understand exactly what teaching computational thinking at the junior level entails, confusion will remain. As Kereluik et al. (2013) points out: "knowing the technology is important, but knowing when and why to use it is more important" (p. 133). One participant demonstrated that computational thinking can be taught in junior classrooms with existing resources such as grids, hopscotch, mazes, or puzzles.

Limitations

As with all research, this investigation has limitations. Because the research involved a small sample of seven participants, it can't be considered to be representative of junior teachers' perceptions and experiences more broadly in New Zealand. However, because it was an exploratory project, it does represent an important first step—currently lacking in the literature—in understanding the experiences of a small group of junior teachers involved in implementing computational thinking. It also highlights areas for future research, including a need to investigate junior teachers' knowledge of computational thinking, junior teachers' knowledge of relevant and useful resources, and how junior teachers' self-efficacy for teaching computational thinking influences decisions about what it is and how it is taught in the classroom. This research is a starting point for that journey.

Conclusion

This exploratory research has uncovered a range of factors that influence the preparedness of these junior teachers to teach computational thinking. Perceived benefits and challenges, knowledge of resources, professional development, support, time constraints, and alignment with assessment are all factors that influence their readiness for teaching the new curriculum content.

Computational-thinking knowledge varied across the participants. Although many teachers were unaware of how much they knew about computational thinking, many believed that all computational-thinking instruction required considerable use of computer-based technology. Perceptions of the benefits of teaching computational thinking, and positive attitudes towards the new curriculum, were greatest when teachers had undertaken professional development and had started to teach computational thinking in their class.

This research has shown that knowing how to use available resources is as influential as the lack of resources. Many teachers knew of resources they would like to have, but didn't know how to use what was already available. An example of this related to assessment. Although computational-thinking assessment is available for New Zealand teachers as part of the new curriculum, this study showed teachers were unaware of these resources. Even teachers who had completed self-directed professional development had not included assessment in their exploration and were unaware of any appropriate assessment. Time was another frequently cited factor driving decisions about learning about computational thinking, and whether to include it in the class timetable.

Comprehensive professional development is needed to ensure misconceptions about computational thinking are addressed and appropriate assessment tools are identified. Due to time constraints, professional development will need to be integrated in the school day (i.e., teachers are released from teaching commitments) to highlight the importance of the new curriculum content.

This research is a starting point for continued research into the successful implementation of computational thinking in junior classrooms.

References

Agarwal, R., Ahuja, M., Carter, P. E., & Gans, M. (1998). *Early and late adopters of IT innovations: Extensions to innovation diffusion theory*. https://www.researchgate.net/publication/228395578_Early_and_late_adopters_of_IT_innovations_Extensions_to_innovation_diffusion_theory

Baylor, A. L., & Ritchie, D. (2002). What factors facilitate teacher skill, teacher morale, and perceived student learning in technology-using classrooms? *Computers & Education, 39*(4), 395–414. https://doi.org/10.1016/S0360-1315(02)00075-1

Bell, J., & Bell, T. (2018). Integrating computational thinking with a music education context. *Informatics in Education*, 17(2), 151–166. doi:10.15388/infedu.2018.09

Bell, T. (2018). Computer science in K–12 education: The big picture. *Olympiads in Informatics, 12*, 3–11. https://doi.org/10.15388/ioi.2018.01

Bell, T., Alexander, J., Freeman, I., & Grimley, M. (2009). Computer science unplugged: School students doing real computing without computers. *New Zealand Journal of Applied Computing & Information Technology, 13*(1), 20–29.

Bell, T., & Roberts, J. (2016). Computational thinking is more about humans than computers. https://www.nzcer.org.nz/nzcerpress/set/articles/computational-thinking-more-about-humans-computers

Blundell, C., Lee, K. T., & Nykvist, S. (2016). Digital learning in schools: Conceptualizing the challenges and influences on teacher practice. *Journal of Information Technology Education: Research, 15*, 535–560. https://doi.org/10.28945/3578

Bolstad, R., Gilbert, J., McDowall, S., Bull, A., Hipkins, R., & Boyd, S. (2012). *Supporting future-oriented learning and teaching: A New Zealand perspective.* Report prepared for the Ministry of Education. Wellington: New Zealand Council for Educational Research and Ministry of Education.

Burnett, C., Dickinson, P., Myers, J., & Merchant, G. (2006). Digital connections: Transforming literacy in the primary school. *Cambridge Journal of Education, 36*(1), 11–29. https://doi.org/10.1080/03057640500491120

Caeli, E. N., & Yadav, A. (2020). *Unplugged approaches to computational thinking: A historical perspective. TechTrends: Linking Research and Practice to Improve Learning. A publication of the Association for Educational Communications & Technology, 64*(1), 29. https://doi.org/10.1007/s11528-019-00410-5

Creswell, J. W., & Plano Clark, V. L. (2018). *Designing and conducting mixed methods research.* (3rd ed.). Sage.

Cviko, A., Mckenney, S., & Voogt, J. (2012). Teachers enacting a technology-rich curriculum for emergent literacy. *Educational Technology Research and Development, 60*(1), 31–54. https://doi.org/10.1007/s11423-011-9208-3

Ertmer, P. A., & Ottenbreit-Leftwich, A. T. (2010). Teacher technology change: How knowledge, confidence, beliefs, and culture intersect. *Journal of Research on Technology in Education, 42*(3), 255–284. https://doi.org/10.1080/15391523.2010.10782551

Ertmer, P. A., & Ottenbreit-Leftwich, A. (2013). Removing obstacles to the pedagogical changes required by Jonassen's vision of authentic technology-enabled learning. *Computers and Education, 64*, 175–182. https://doi.org/10.1016/j.compedu.2012.10.008

Falloon, G. (2016). An analysis of young students' thinking when completing basic coding tasks using Scratch Jnr. on the iPad. *Journal of Computer Assisted Learning, 32*(6), 576–593.

Falloon, G. (2019). Using simulations to teach young students science concepts: An experiential learning theoretical analysis. *Computers & Education, 135*, 138–159. https://doi.org/10.1016/j.compedu.2019.03.001

Gibson, S. E., & Brooks, C. (2012). Teachers' perspectives on the effectiveness of a locally planned professional development program for implementing new curriculum. *Teacher Development, 16*(1), 1–23.

Giffen Cheng, Y. H., Chou, W. S., & Cheng, C. W. (2014). Research into teachers' receptivity for arts infused curricula in Taiwan. *International Journal of Art & Design Education, 33*(1), 55–74. https://doi.org/10.1111/j.1476-8070.2014.12035.x

Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the field. *Educational Researcher, 42*(1), 38–43. https://doi.org/10.3102/0013189X12463051

Hattie, J., & Zierer, K. (2017). 10 mindframes for visible learning: Teaching for success. Routledge.

Hunter, P. A., & Keown, P. A. (2001). The New Zealand social studies curriculum struggle 1993–1997: An "insider" analysis. *Waikato Journal of Education, 7*, 55–72. https://hdl.handle.net/10289/6268

Kereluik, K., Mishra, P., Fahnoe, C., & Terry, L. (2013). What knowledge is of most worth: Teacher knowledge for 21st century learning. *Journal of Digital Learning in Teacher Education, 29*(4), 127–140. https://doi.org/10.1080/21532974.2013.10784716

Kia Takatū ā-Matihiko. (n.d.). *Kia Takatū ā-Matihiko: Digital Readiness.* https://technology.tki.org.nz/Teacher-education/Technology-Online-webinar-recordings/Kia-Takatu-a-Matihiko-Digital-Readiness

Massey University. (2018). *Massey University code of ethical conduct for research, teaching and evaluations involving human participants*. Massey University.

Mind Lab. (n.d.). Digital passport: The solution to understanding the new digital curriculum. https://www.digitalpassport.co.nz/

Ministry of Education. (2017). *The New Zealand Curriculum Online: Technology*. http://nzcurriculum.tki.org.nz/The-New-Zealand-Curriculum/Technology

Ministry of Education. (n.d.-a). *Computational thinking progress outcomes.* Retrieved from http://technology.tki.org.nz/Technology-in-the-NZC/CT-Progress- outcomes-exemplars-and-snapshots

Ministry of Education. (n.d.-b). Indicators of progression: NZC components (levels 1–8). https://technology.tki.org.nz/Technology-in-the-NZC/Technology-indicators

Moats, L. C. (2010). Speech to print: Language essentials for teachers. (2nd ed.). Paul H. Brookes.

Moltó, O., Domingo, L., & Gil, J. M. S. (2009). Digital learning, analogical schools. *International Journal of Learning: Annual Review, 16*(11), 293–304. https://doi.org/10.18848/1447-9494/CGP/v16i11/46732

Nouri, J., Zhang, L., Mannila, L., & Norén, E. (2020). Development of computational thinking, digital competence and 21st century skills when learning programming in K–9. *Education Inquiry, 11*(1), 1–17. https://doi.org/10.1080/20004508.2019.1627844

Rogers, E. M. (2010). Diffusion of innovations. (4th ed.). https://books.google.co.nz/books?id=v1ii4QsB7jIC

Sotiriou, S., Riviou, K., Cherouvis, S., Chelioti, E., & Bogner, F. X. (2016). Introducing large-scale innovation in schools. *Journal of Science Education and Technology, 25*(4), 541–549. https://doi.org/10.1007/s10956-016-9611-y

Tahi Rua Toru Tech. (2018). *Tahi rua toru tech*. https://123tech.nz/

Tell, C. A., Bodone, F. M., & Addie, K. L. (2000). *A framework of teacher knowledge and skills necessary in a standards-based system: Lessons from high school and university faculty*. Paper presented at the Annual Meeting of the American Educational Research Association (New Orleans, LA, April 24–28, 2000). https://eric.ed.gov/?id=ED447206

Terrazas-Arellanes, F. E., Knox, C., Strycker, L. A., & Walden, E. (2016). A face-to-face professional development model to enhance teaching of online research strategies. *Journal of Information Technology Education, 15*, 335–367. https://doi.org/10.28945/3536

Vlahu-Gjorgievska, E., Videnovik, M., & Trajkovik, V. (2018). Computational thinking and coding subject in primary schools: Methodological approach based on alternative cooperative and individual learning cycles. In *IEEE International Conference on Teaching, Assessment, and Learning for Engineering (TALE)* (pp. 77–83).

Waugh, R. F. (2000). Towards a model of teacher receptivity to planned system-wide educational change in a centrally controlled system. *Journal of Educational Administration, 38*(4), 350–367. https://doi.org/10.1108/09578230010373615

Wing, J. M. (2006). Computational thinking. *Communications of the ACM, 49*(3), 33–35. https://doi.org/10.1145/1118178.1118215

Wing, J. M. (2017). Computational thinking's influence on research and education for all. *Italian Journal of Educational Technology, 25*(2), 7014. https://doi.org/10.17471/2499-4324/922

Yin, H., Lee, J., & Jin, Y. L. (2011). Teacher receptivity to curriculum reform and the need for trust: An exploratory study from southwest China. *Asia-Pacific Education Researcher, 20*(1), 35–47.

Biographical notes

Jeanette Irons
jmmirons@gmail.com

Jeanette Irons has a Master of Education in E-Learning, and is an experienced primary school teacher. She is a marking assistant for Massey University's postgraduate digital education programme.

Maggie Hartnett
M.Hartnett@massey.ac.nz

Dr Maggie Hartnett is a senior lecturer at Massey University where she teaches digital education. She is programme coordinator for the postgraduate digital education programmes and is associate editor of the *Journal of Open, Flexible and Distance Learning*. Her research interests include motivation and engagement in digital environments, digital inclusion, teaching and learning with digital technologies, learner support, and digital spaces for learning.

Irons, J., & Hartnett, M. (2020). Computational thinking in junior classrooms in New Zealand. *Journal of Open, Flexible and Distance Learning, 24*(2), [28–42.].