Chang, M. et al. (Eds.) (2019). Proceedings of the 27th International Conference on Computers in Education. Taiwan: Asia-Pacific Society for Computers in Education

Computational Thinking Development Challenges: Case Studies In Thai Primary Education

Kantinee KATCHAPAKIRIN *, Chutiporn ANUTARIYA
School of Engineering and Technology, Asian Institute of Technology, Thailand
*st120200@ait.ac.th, chutiporn@ait.ac.th

Abstract: Computational Thinking (CT) has been introduced in the Thai standard curriculum since 2017. In order to effectively support implementation of computing education, it is necessary to understand traditional classroom practices as well as challenges that teachers experience in CT development and assessment. The identification of challenges allows researchers to propose relevant learning enhanced technology to efficiently develop students' CT skill. This study explored practical challenges and their importance in Grade 4th and 5th classrooms in Thai schools by conducting a preliminary survey including case studies and a poll survey. The case studies were carried out to uncover the outlooks and perspectives of teachers on challenges and their current strategies to alleviate those challenges. The poll survey was conducted to measure and identify the top five challenges from the teachers' perspective including: less time in the curriculum, lack of material, limited fundamental knowledge, lack of confidence and student differentiation. Practical strategies adopted in Thai CT Classroom were explored, and relevant learning enhanced technology was recommended for future study, which could be beneficial for researchers and responsible agencies to significantly improve the quality of CT education in Thailand.

Keywords: Computational Thinking, Challenges, Strategies, Thailand Curriculum, Computing Science, Primary education, computer science education; K-12 CS curriculum.

1. Introduction

Computational Thinking (CT) has increasingly become important in all areas of society (Heintz, 2016) and has been considered as a social consensus skill related to problem-solving, systematic and algorithmic thinking (Haseski, 2018). It is suggested to be an essential fundamental skill for everyone like reading, writing, and arithmetic skills (Wing, 2017). Grover (2018) also argued that it should be added as the 5th "C" into the "4C skills of the 21st-century framework" and should be taught to all students. Thus, due to its increasing popularity and criticality to the 21st century workforce, CT has been introduced into the basic education curriculum in many countries (Heintz, 2016). Thailand has also shifted its regard for CT from a specialized skill (Ministry of Education, 2008) to a fundamental skill (Ministry of Education, 2017). It is now considered under Science learning area rather than Occupations and Technology learning area in Computing Science.

In Revised Thai Curriculum, Computing Science subject consists of three main strands: Digital Literacy (DL), Information and Communication Technology (ICT), and Computer Science (CS) (IPST, 2017). DL involves wisdom, safety, and ethic ways to use technology safely, respectfully and responsibly. Students would be taught how to evaluate information and consider how reliable the information is. ICT involves how to use the computers and their applications as tools to create, organize, analyze, and visualize data or other digital content to support decision making. CS emphasizes on thinking, i.e., computational thinking which aims to foster students to be able to solve problems using logics and algorithms. Students should understand the fundamental CS principles and concepts, be able to analyze problems in computational ways, and have practical programming experience to cooperate with computers to solve problems. The implementation of the revised curriculum takes 3 years of roll-out plan. In the academic year 2018, the curriculum was applied for

Grade 1st and Grade 4th, then applied for Grade 2nd and Grade 5th in 2019, and as of the academic year 2020, the curriculum shall be applied for all grades.

To efficiently improve the quality of computing science education, it is necessary to understand how computational thinking was taught in Thai classrooms and what challenges teachers experienced and how they dealt with the challenges. The identification of such challenges could enable proper employment of relevant Learning Enhanced Technology to efficiently develop students' computational thinking skills and to further elevate the Computational Thinking agenda in Thailand education.

This paper reviews existing research in Section 2. Then, Sections 3 and 4 discuss a preliminary field study to elicit practical challenges and their importance in the Grades 4th – 5th of computing science classrooms in Thai schools, and to analyze current pedagogical strategies that Thai teachers adopted. Finally, Section 5 concludes and suggests potential Learning Enhanced Technology research to overcome those important challenges.

2. Related works

2.1 CT Development and Assessment Challenges

This sub-section reviews existing studies related to the challenges that teachers experienced in computing classroom, categorized into three aspects: *teachers*, *students*, and *environment*. We further applied the work of Finger and Houguet (2009) to classify the teacher's and student's challenges into *intrinsic* and *extrinsic* challenges. The former ones are challenges that teachers and students can fully manage by their own while the latter ones are those that they have less control. Table 2 discusses the teacher's challenges, whereas Table 2 and Table 3 respectively summarize student's challenges and environment-related challenges, which can be categorized into material, curriculum, infrastructure and other subcategories.

Table 2

Teacher's Intrinsic and Extrinsic Challenges

Teacher's Challenge	**Description**
Intrinsic Challenge	
Limited Fundamental Knowledge	Teachers have little subject knowledge and skills. They lack understanding in digital literacy, programming, and troubleshooting (Sentance, 2017). Some teachers misunderstand CT concepts and falsely described CT as the basic use of computers or technology (Ribeiro, 2013; Lockwood, 2017; Bower, 2015; Mouza, 2017; Yadav, 2014; Maruyama, 2017). Educating teachers with relevant knowledge and skills has been a great barrier in many country's education (Heintz, 2016).
Limited Pedagogical Knowledge	Teachers have limited teaching approaches to make Computational Thinking concepts accessible, engaging, interesting and fun (Sentance, 2017; Silapachote, 2018; Black et al., 2013).
Lack of Experience	Teachers learned the topics by themselves. So, they have no experience to bring the topics to life (Lockwood, 2017; Sentance, 2017).
Lack of Confidence	Teachers are concerned about their own knowledge in Computer Science and programming (Lockwood, 2017; Sentance, 2017).
Lack of Time Management	Teachers lack preparation time to refine lesson plans and develop competency in the material (Cho, 2014; Lockwood, 2017; Sentance, 2017).
Extrinsic Challenge	
Less Popularity of ICT Teachers	Lack of trained ICT teachers who have the knowledge and skills to embed CT in school curricula (Barr, 2011).
Low Quality of Building CS Educators	Current undergraduate courses focus on teaching computer science student to use computers and does not consider much about CT. (Ribeiro, 2013).

Table 3

Student's Challenges

Student's Challenge	Description
Intrinsic Challenge	
Limited Computing Literacy Skills	Students cannot memorize common computing keywords. So, they could not read code, reuse code (Basu, 2016) or detect errors (i.e. grammatical, logical or syntactic errors) (Sentance, 2017) that exist within that code.
Lack of Understanding of Topics	Even though students can memorize computing keywords, but they do not understand why and how and when the keywords were used. They had difficulties in understanding the meaning and use of variables, abstraction, functions, conditionals and loops (Basu, 2016; Sentance, 2017; Grover, Pea, & Cooper, 2015).
Low Mathematic Ability	Students lack capability and competence in mathematical concepts e.g. Boolean Algebra, Logical Operators and manipulating numbers (Sentance, 2017).
Limited Problem-solving Skills	Students can memorize and understand the computing concepts, but they cannot apply the knowledge to new problems or other subjects (Sentance, 2017). Students lack abstraction, modeling, debugging skills while they were reading, analyzing and synthesizing problems in order to abstract the essential data to solve a problem (Basu, 2016).
Low Students' Resilience	Students have no resilience to keep trying when something does not work and give up easily. (Sentance, 2017; Mooney, 2014).
Low Engagement and Motivation	Students have low interest and cannot maintain their attention to engage in the classroom (Grover, 2019; Sentance, 2017). Moreover, they do not develop their mastery of the Computing subject outside of school hours (Mooney, 2014).
Extrinsic Challenge	
Differentiation	Students were different. They had different skill and experience in programming. They learned at different pace and learning styles. Computing subject had the most widening gap between students greater than other subjects (Sentance, 2017). The challenge of Differentiation was classified as Students' extrinsic challenge in this study rather than teachers' intrinsic in Sentence work (Sentance, 2017) because it related to students and students have less control on it. However, the extrinsic challenge of one could possibly be intrinsic of others.

2.2 Strategies to overcome the challenges

This sub-section reviews strategies that have been used to overcome the challenges as reported in the literature.

Teacher Workshop. Lockwood (2017) informed that teacher workshops are extremely effective strategies at equipping teachers with relevant subject knowledge and skills and changing teachers' perceptions. Moreover, days-long practical sessions are hugely beneficial to give teachers the confidence and experience to introduce the lessons into their classes. (Falkner, 2015; Morreale, 2012; Pokorny, 2012; Yadav, 2014; Bargury, 2012; Carvalho, 2013). Heintz (2016) reported that a common struggle among countries is training of teachers. Another strategy was proposed, *Online Learning MOOC*. With appropriate MOOC support, Australia and Norway can introduce CT into their classrooms in short time period (Vivian , 2014; Falkner, 2015). However, this strategy mainly overcome only the challenges of *Limited Fundamental Knowledge* and *Limited Pedagogical Knowledge* and cannot reduce the impact of *Lack of Confidence*. *Teacher Network and Community* reportedly has positive impact on teacher's confidence, energetic. For example, Computing at school (CAS) community in England supports teachers to share their teaching ideas and allows experienced teachers to support other teachers. Heintz (2016) stated that participated teachers gain more confidence and the number of isolated teachers is reduced. Falkner (2015) used a Google+ group community to foster collaboration of the MOOC.

Suitable teaching materials were very important for teachers. Many countries developed *Teaching Material repository* as centralized national resources (Heintz, 2016). In New Zealand, they launched "CS Field Guide" as their national free online open-source teaching material repository. Similarly, Norway has started Laer Kidsa Koding (Http://kidsakoder.no/) to provide extensive teaching

material for teachers. In addition, *Contextualization of learning* is to design material that relate computing to other subjects or other real-world activities can support better students' learning (Sentance, 2017) and possibly lessen student *Limited Problem-Solving Skills* challenges.

CT skills can be taught ranging from low level to high level i.e. *Unplugged activities to Practical computerized activities*. A significant proportion of teachers adopt the unplugged approach to support students' understanding in the classroom (Sentence, 2017; Lockwood, 2017). Shuchi (Grover , 2019) suggested using unplugged activities before working in programming contexts in environments like Scratch. However, coding hands-on is an approach in which CT skills can be simulated and evaluated as a student's ability to program a solution to a problem. Using demonstrations, learning interactive lessons, learning through videos are also strategies that teachers suggest (Lockwood, 2017).

Table 4

Environment-Related Challenges

Environment -related Challenge	**Description**
Material	
Lack of Material	Lack of good quality or age- or grade-appropriate material also poses an environmental challenge. There is also lack of material to contextualize CT other disciplines (Sentance, 2017). Heintz (2016) also support that developing suitable teaching material is a common challenge on introducing CT to all countries.
Curriculum	
Less Time in the Curriculum	School hours were not enough to achieve the expectation of the curriculum.
Lack of CT Assessment Guidelines in the Curriculum	There is little guidance on assessment that can be applied in practical. As the assessment of CT is complicated, assessing the progression of students following instructions was not the mechanism for assessing CT. Teachers should formatively assess students and prepare students for summative assessment tasks (Sentance, 2017). Many researchers (Van, 2001; Roman, 2017; Tissenbaum, 2018) proposed a diversity of CT assessment methods which poses challenges about the appropriate methodology for assessing CT learning in practices.
Infrastructure	
Lack of Resources	Hardware, software resources (e.g. Computers) and Internet services are not adequate to teach (Sentance, 2017; Ribeiro, 2013). Carvalho (2013) reported the poor conditions of infrastructure in Brazilian public schools was an impediment to develop CT skills in computerize activities.
Technical Difficulties in School	There are some technical problems with getting software to work on the school network such as software installation problem, network problem and security problem (Sentance, 2017).
Lack of Support from IT Departments	Unwillingness of technician support is also a great challenge for developing CT (Lockwood, 2017). Sentance (2017) reported that their technician support considers the software may break the integrity and security of the school's computer network therefore they are reluctant to maintain and troubleshoot installed software.
Other	
Parents' Attitude	Parents' attitude toward CT development in primary school influence on children's attitude (Maruyama, 2017).

Since programming is the most challenging part of computer science in schools. There were several interesting strategies proposed to support students. *Collaborative working*. There were a variety of collaborative working strategies that teachers used within the classroom such as peer mentor, pair programming, teamwork and collaboration. Kafai and Burke (2014) defined it as a concept of computational participation. In this context, student who are good at working with coding can support others in the classroom. In addition, students can discuss with peers to find out solutions for specific problems (Passey, 2014). As a result, this strategy could minimize the *differentiation* in students' ability. *Scaffolding programming tasks*. A range of activities supporting students' scaffolding

programming tasks have been proposed; Code walkthroughs (Van Gorp and Grissom, 2001), reading and tracing code (Lopez et al., 2008), use trace tables to help students understand the flow of control and the changing value of variables with in a program, extend and debug code (Sentance, 2017). Lister (Lister, 2011) argued that students need to be able to trace code with greater than 50% accuracy before starting to write their own program. Moreover, writing algorithms in groups, and inserting comments in pairs were suggested by Van and Grissom (Van Gorp and Grissom, 2001) *Classroom teaching assistant*. Cho et al. (Cho et al., 2014) states that a significant proportion of teachers identified the desire of a classroom assistants. Basu (Basu, 2016) identified that the human-provided-scaffolds minimizes the challenges students face over learning time period.

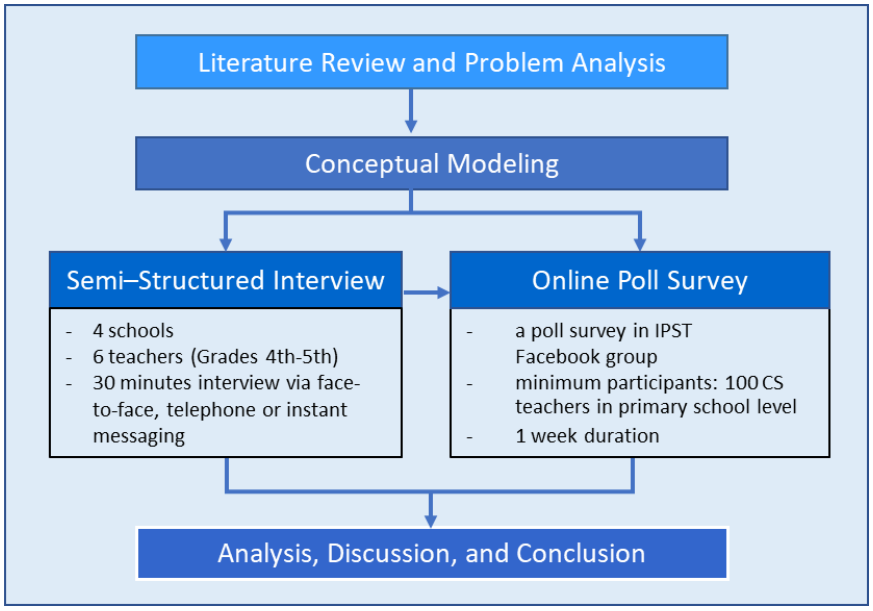

Figure 6. Overview of research methodology.

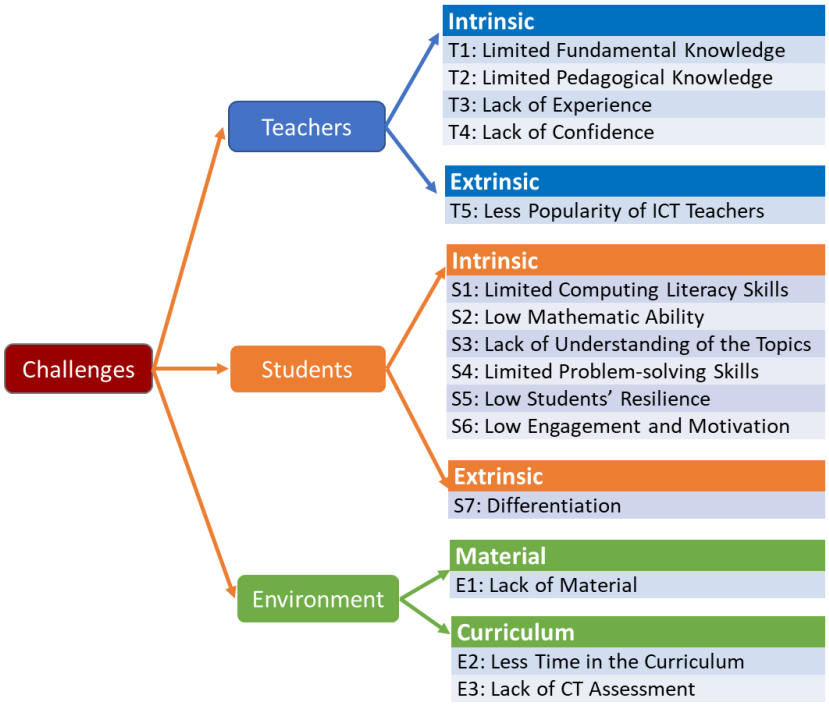

Figure 7. The proposed challenge model.

3. Research Methodology

This study aimed to gain insights into the challenges experienced by teachers during the implementation of computational thinking development within primary school settings in Thailand as well as strategies Thai teachers adopted to deal with those challenges. Given these considerations, **錯誤! 找不到參照來源。** illustrates the proposed research methodology, which initially started by reviewing existing related research and analyzing the problem. Then, a conceptual model defining relevant challenges in Thailand was preliminarily analyzed and developed which comprises 5 teacher's challenges, 7 student's challenges and 3 environment-related challenges (cf. **錯誤! 找不到參照來源。**). Next, semi-structured interviews and online poll survey were conducted with in-service computing science teachers to verify the defined challenges and findings. Lastly, the results were analyzed, and conclusions drawn. Sub-sections 3.1 and 3.2 explains the design of the semi-structured interviews and poll survey in details.

3.1 Semi-structured Interview Design

Semi-structured interviews were used as a tool to verify the proposed conceptual model, which defines important challenges involving CT development and assessment in Computing Science subject, and comprise the following primary questions:

(1) How do you support students to develop their Computational Thinking?
(2) How can you assess Computational Thinking skills of an individual student?
(3) What are important challenges in developing Computational Thinking from teachers' view?
(4) How have you dealt with those challenges?

Table 5 gives an overview of each teacher and his/her school who agreed to participate in the interview. Note that all teachers instructed CT in computing science classes of Grade 4th and Grade 5th and were from four different schools having different sizes and located in different provinces. Section 4 discusses the results and findings obtained from the interview.

Table 5

An Overview of Schools and Teachers Participating in the Study

School	Province	Size	Host	Teachers	Level	Students per Class	CT Tools
1. Provincial#1	Phetchaburi	Medium	OBEC	1. Tommy	G4-G6	40	Scratch
2. Provincial#2	Roi-Ed	Small	OBEC	2. Lilly	G2-G6	<10	Code.org, Robotic, Microbit
3. Bangkok#1	Bangkok	Extra Large	BMA	3. Susan	G4	33	Scratch
				4. Jenny	G5	35	Scratch
4. Bangkok#2	Bangkok	Extra Large	BMA	5. Grace	G4	45	Scratch
				6. Katie	G5	45	Scratch

3.2 Poll Survey Design

In order to generalize and measure the importance of the specified challenges from teachers' perspectives, a poll survey was conducted online in a formal CS primary level teacher community. The community has been established by the Institute for the Promotion of Teaching Science and Technology (IPST) for sharing computing science resources among teachers. The poll asked the participating teachers to identify top three challenges faced while teaching CT using Scratch for Grade 4th and Grade 5th classrooms. The number of participating teachers were around 100.

Table 5

Interview and Poll Results: Challenges on CT Development in Thai CS Classroom

	Challenges Proposed in the Conceptual model													
	T1	**T2**	**T3**	**T4**	**T5**	**S1**	**S2**	**S3**	**S4**	**S5**	**S6**	**S7**	**E1**	**E2**	**E3**
Interview Results:															
- Number of teachers agreed that the challenges had high impact	5		2	4		6	6	6	6			6	6	4	
Online Poll Results:															
- Number of votes from online poll	49	20		49	19						20	43	83	93	10
- Poll ranking	3			3								5	2	1	
High Impact Challenges	×			×		×	×	×	×			×	×	×	

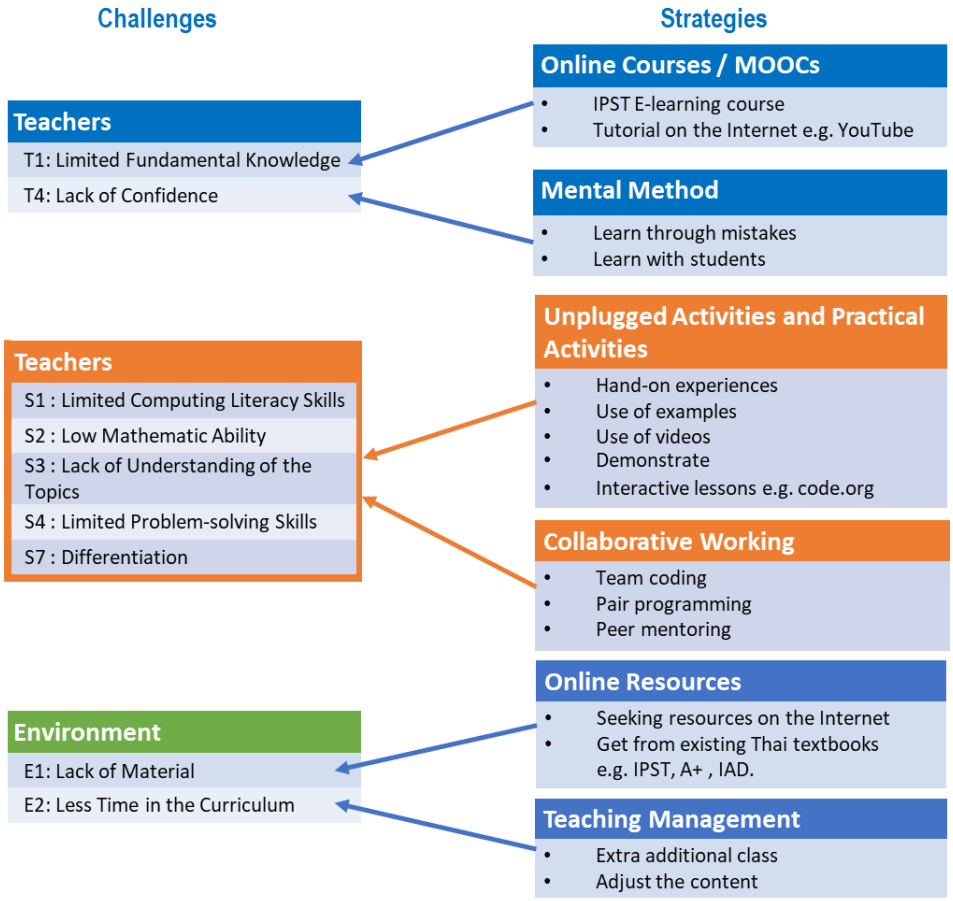

Figure 8. Practical strategies for the important challenges in Thai CT Classroom.

4. Findings and Discussion

The traditional practices for computing science classroom in four Thai schools followed the common sequence of lecturing, demonstration, practice, and evaluation. The process started by teachers giving lectures on CT concepts followed by demonstration of exercises. Students were then allowed to practice exercises either individually or in pair. This whole process was carried out within 30 minutes – 2 hours per week. The level of CT instruction and the level of contextualization varied among schools depending on the different levels of readiness of teachers and schools. The school *Provincial#2*, is an example of a well-equipped school with both teacher's qualification and school environment. Lilly

adopted physical programmable devices and integrated CT with STEM subjects, while the school *Bangkok#1* mostly taught CT using unplugged activities.

Table 5 summarizes the number of teachers who participated in the interview and polls and agreed that for each challenge, it had high impact to the teaching and learning as proposed in the conceptual model. Thus, the challenges supported by all six interviewees and or appeared in the top five poll ranking were selected as the important challenges with high impact. **錯誤! 找不到參照來源。** then reported the strategies that the teachers employed to deal with the challenges. For instance, to handle the teacher's challenge with *limited fundamental knowledge*, the teachers referred to online courses and MOOCs as their practical solutions. Likewise, to overcome the *lack of confidence* challenge, the teachers applied an open-minded approach by accepting mistakes and learning along with the students.

The important challenges related to students included *differentiation*, *limited computing literacy skills*, *low mathematic ability*, *lack of understanding of the topics* and *limited problem-solving skills*. The teachers applied two strategies: (i) unplugged and practical activities such as hand-on experiences, examples, video, demonstrate and interactive lesson, and (ii) collaborative working such as team coding, pair programming and peer mentoring.

For the challenges related to learning environment, *less time in the curriculum* and *lack of material* were discussed as the most important challenges. Several approaches were employed to deal with the first challenge. Interesting ones include (i) spending less time on lecturing and increase more time on experimenting, and hence encouraging students to practice more under their supervision, (ii) adding extra instruction time to cover all contents, (iii) adjusting the contents based on the available time. To cope with the lack of material challenge, searching and reusing resources from the Internet along with the information from available textbooks were adopted as a solution.

Although the CT assessment (i.e., *lack of CT assessment*) was not identified by the conducted survey as an important challenge, it is a central to support students to develop their CT skills (Grover, 2017). This study found that the primary assessment method used in most schools involved evaluating the correctness of the final output. However, Grover (2017) suggested that evaluating simply the final results and disregarding the learning process observation would not support proper intervention and knowledge mastery when applying CT to other problem domains. Only the school *Provincial#2* reported the use of *observation* as an assessment method and the capability to give immediate feedback to students individually. While other teachers reported that it was impossible to have formative assessment and individual feedback due to the class size. Therefore, the bigger the class size, the higher the complexity of CT development and assessment. In addition, to evaluate the students' problem-solving skill, incorrect codes were given to them to debug and correct. It is also found that besides showing the results on computer screen, coding a real, physical robot could better improve student understanding, engagement and motivation. This was evidenced by the school *Provincial#2*, where students paid high attention to study STEM subjects under CT environment.

5. Conclusion and Recommendations

This study aimed to understand CT development challenges that teachers experienced in traditional classroom practices. The challenges in the literature were explored and categorized into three aspects: teachers, students, and environment. Then, we analyzed challenges with the context of the Thai CT classroom. Totally fifteen challenges were selected and included in the designed conceptual model. We validated the conceptual model by conducting semi-structured interviews and a poll survey. The clear set of nine significant challenges on CT development from teachers' perspective were identified: T1: Limited Fundamental Knowledge, T4: Lack of Confidence, S1: Limited Computing Literacy Skills, S2: Low Mathematic Ability, S3: Lack of Understanding of the Topics, S4: Limited Problem-solving Skills, S7: Differentiation, E1: Lack of Material ,and E2: Less Time in the Curriculum. While Less Time in the Curriculum and Lack of CT Assessment were the top two common challenges in the poll survey. In additional, the study found that teaching management, self-learning through online resources and collaborative working are strategies that Thai teachers usually used to overcome the challenges.

To reduce the impact of the challenges, the following relevant research in the area of learning enhanced technology is proposed here as a potential scalable strategy to effectively develop CT skills in students, and thus improving the quality of CT education in Thailand:

- Adaptive and personalized system, which aims to customize learning for each student's strengths, needs, skills and interests in order to support a diversity of students in various dimensions such as learning skills, learning styles, and learning pace.
- CT assessment techniques are also required for efficient CT intervention. The techniques evaluate the process of thinking rather than evaluating the correctness of the final outputs.
- Conversational agent, intelligent agents, computational linguistics, NLP techniques could possibly support students for self-practice out of school hours with proper guidance, and also support teachers for assessing CT skills accurately.
- Digital repository and material recommender system are required to archive and intelligently recommend appropriate learning materials for each student.
- Gamification techniques can benefit student engagement and motivation.

Acknowledgments

This research was supported in part by Thailand Science Research and Innovation (Grant Number RDG6240046).

References

Anderson, R. E. (1992). Social impacts of computing: Codes of professional ethics. *Social Science Computing Review*, *10*(2), 453-469.

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, 2(1), 48-54. https://doi.org/10.1145/1929887.1929905

Basu, S., Biswas, G., Sengupta, P., Dickes, A., Kinnebrew, J. S., & Clark, D. (2016). Identifying middle school students' challenges in computational thinking-based science learning. Research and practice in technology enhanced learning, 11(1), 13.

Black, J., Brodie, J., Curzon, P., Myketiak, C., McOwan, P. W., & Meagher, L. R. (2013). Making computing interesting to school students: teachers' perspectives. In Proceedings of the 18th ACM conference on Innovation and technology in computer science education, (pp. 255-260): ACM.

Bower, M., & Falkner, K. (2015). Computational thinking, the notional machine, pre-service teachers, and research opportunities. Proceedings of the 17th Australasian Computer Education Conference (ACE2015) (pp.37-46), Sydney, Australia.

Carvalho, T., Andrade, D., Silveira, J., Auler, V., Cavalheiro, S., Aguiar, M., ... & Reiser, R. (2013, October). Discussing the challenges related to deployment of computational thinking in brazilian basic education. In 2013 2nd Workshop-School on Theoretical Computer Science (pp. 111-115). IEEE.

Cho, S., Pauca, P., & Johnson, D. (2014, March). Computational thinking for the rest of us: A liberal arts approach to engaging middle and high school teachers with computer science students. In Society for Information Technology & Teacher Education International Conference (pp. 79-86). Association for the Advancement of Computing in Education (AACE).

Falkner, K., Vivian, R., & Falkner, N. (2015, January). Teaching computational thinking in K-6: The CSER digital technologies MOOC. In Proceedings of the 17th Australasian Computing Education Conference (ACE 2015) (Vol. 27, p. 30).

Finger, G., & Houguet, B. (2009). Insights into the intrinsic and extrinsic challenges for implementing technology education: Case studies of Queensland teachers. International Journal of Technology and Design Education, 19(3), 309-334.

Grover, S., Pea, R., & Cooper, S. (2015). Designing for deeper learning in a blended computer science course for middle school students. Computer Science Education, 25(2), 199-237.

Grover, S. (2017). Assessing algorithmic and computational thinking in K-12: Lessons from a middle school classroom. In Emerging research, practice, and policy on computational thinking (pp. 269-288). Springer, Cham.

Grover, S. (2018, February 25). The 5th 'C' of 21st Century Skills? Try Computational Thinking (Not Coding). Retrieved from Edsurge: https://www.edsurge.com/news/2018-02-25-the-5th-c-of-21st-century-skills-try-computational-thinking-not-coding

Grover, S., Lundh, P., & Jackiw, N. (2019, February). Non-programming activities for engagement with foundational concepts in introductory programming. In Proceedings of the 50th ACM Technical Symposium on Computer Science Education (pp. 1136-1142). ACM.

Haseski, H. I., Ilic, U., & Tugtekin, U. (2018). Defining a New 21st Century Skill-Computational Thinking: Concepts and Trends. International Education Studies, 11(4), 29-42.
Heintz, F., Mannila, L., & Färnqvist, T. (2016, October). A review of models for introducing computational thinking, computer science and computing in K-12 education. In 2016 IEEE Frontiers in Education conference (FIE) (pp. 1-9). IEEE.
IPST. (2017). Computing Science Teacher Guides. Thailand: The Institute for the Promotion of Teaching Science and Technology.
Kafai, Y. B., & Burke, Q. (2014). Connected Code: Why Children Need to Learn Programming. MacArthur Foundation Series on Digital Media and Learning. MIT Press.
Lockwood, J., & Mooney, A. (2017). Computational Thinking in Education: Where does it fit? A systematic literary review. arXiv preprint arXiv:1703.07659.
Maruyama, Y., Kanoh, H., & Adachi, K. (2017). A Preliminary Investigation into Parents' Concerns about Programming Education in Japanese Primary Schools. International Association for Development of the Information Society.
Ministry of Education. (2008). The Basic Education Core Curriculum BE 2551 (AD 2008). Kurusapa Ladprao Publishing.
Ministry of Education. (2017). Indicators and content areas in science (revised curriculum A.D. 2017) according to Basic Education Core Curriculum B.E. 2551 (A.D. 2008). Bangkok, Thailand: The Agricultural Co-operative Federation of Thailand Press.
Mooney, A., Duffin, J., Naughton, T. J., Monahan, R., Power, J. F., & Maguire, P. (2014). PACT: An initiative to introduce computational thinking to second-level education in Ireland.
Morreale, P., Goski, C., Jimenez, L., & Stewart-Gardiner, C. (2012). Measuring the impact of computational thinking workshops on high school teachers. Journal of Computing Sciences in Colleges, 27(6), 151-157.
Mouza, C., Yang, H., Pan, Y.-C., Yilmaz Ozden, S., & Pollock, L. (2017 in press). Resetting educational technology coursework for pre-service teachers: A computational thinking approach to the development of technological pedagogical content knowledge (TPACK). Australasian Journal of Educational Technology. https://doi.org/10.14742/ajet.3521
Passey, D. (2014). Intergenerational learning practices—Digital leaders in schools. Education and Information Technologies, 19(3), 473-494.
Pokorny, K. L., & White, N. (2012). Computational thinking outreach: reaching across the K-12 curriculum. Journal of Computing Sciences in Colleges, 27(5), 234-242.
Ribeiro, L., Nunes, D. J., da Cruz, M. K., & de Souza Matos, E. (2013, October). Computational thinking: Possibilities and challenges. In 2013 2nd Workshop-School on Theoretical Computer Science (pp. 22-25). IEEE.
Román-González, M., Moreno-León, J., & Robles, G. (2017, July). Complementary tools for computational thinking assessment. In Proceedings of International Conference on Computational Thinking Education (CTE 2017), S. C Kong, J Sheldon, and K. Y Li (Eds.). The Education University of Hong Kong (pp. 154-159).
Sentance, S., & Csizmadia, A. (2017). Computing in the curriculum: Challenges and strategies from a teacher's perspective. Education and Information Technologies, 22(2)
Silapachote, P., & Srisuphab, A. (2017). Engineering Courses on Computational Thinking Through Solving Problems in Artificial Intelligence. International Journal of Engineering Pedagogy, 7(3), 34-49.
Tissenbaum, M., Sheldon, J., Sherman, M. A., Abelson, H., Weintrop, D., Jona, K., ... & Snow, E. (2018). The state of the field in computational thinking assessment. International Society of the Learning Sciences, Inc.[ISLS]..
VAN Gorp, M. J., & Grissom, S. (2001). An empirical evaluation of using constructive classroom activities to teach introductory programming. Computer Science Education, 11(3), 247–260.
Vivian, R., Falkner, K., & Falkner, N. (2014). Addressing the challenges of a new digital technologies curriculum: MOOCs as a scalable solution for teacher professional development.
Wing, J. M. (2017). Computational thinking's influence on research and education for all. Italian Journal of Educational Technology, 25(2), 7–14.
Yadav, A., Mayfield, C., Zhou, N., Hambrusch, S., & Korb, J. T. (2014). Computational thinking in elementary and secondary teacher education. ACM Transactions on Computing Education (TOCE), 14(1), 5.