Can Secondary School Mathematics Students Be Taught To Think Computationally?

Weng Kin Ho wengkin.ho@nie.edu.sg
Wendy Huang wendy.huang@nie.edu.sg
Chee Kit Looi cheekit.looi@nie.edu.sg
National Institute of Education
Nanyang Technological University
Singapore

Suranteran Longanathan
suranteran_loganathan@moe.edu.sg
Ministry of Education
Singapore

26 July 2018

Abstract

The thrust of this paper is to demonstrate that secondary school mathematics students can indeed be taught to think computationally. We do so by applying APOS theory to design mathematics lessons that intentionally integrate Computational Thinking (CT) and Mathematical Thinking (CT), and implementing these tailor-made lessons in authentic classroom environment of a Singapore secondary school. Based on lesson observation notes and post-lesson discussions with teachers, we present evidence to confirm the hypothesis that integrating CT and MT deepens students' understanding of mathematical concepts.

1 A case to establish

The world in which we live in is continuously shaped by automation and algorithms as computer-based technologies become integrated into every aspects of life – ranging from the smart phone, through web-based classrooms, to driverless automobiles. There is thus an evidently urgent need for all not only to embrace technology as users but also to understand how these technologies function so that as individuals one may make informed decisions to invoke them as well as to guard against their misuse [21, Chapter 4]. To achieve such a level of understanding would entail a unique paradigm which is tuned to the thinking of a computer scientist (or a human programming computers). This paradigm which we aim to expound on herein is *Computational Thinking* (CT, for short) and, as [6] puts, it refers to

> "the thought processes involved in formulating problems and their solutions so that the solutions are represented in a form that can be effectively[1] carried out by an information-processing agent".

Given the rapid advances in Information and Computer Technology as well as the industrial demand for a workforce that is equipped with the needful computational knowledge and skills, it has never been more compelling than now to advocate CT as a foundational literacy. This calls for schools to teach children CT as early as possible in the child's educational development.

Educational systems from many countries respond to this calling at different points along a spectrum of intensities and involvement of teaching CT in schools. On one end of the spectrum, the United Kingdom has fully embarked on the enterprise of teaching Computing as a subject, beginning from Primary School education in 2014. This core subject status was articulated in a 2014 report for Computing At School (CAS) in the UK [13]:

> "Computer science is a proper, rigorous school subject discipline, on a par with mathematics or chemistry, that every child should learn from primary school onwards."

In particular, the disciplinarity of Computer Science requires a distinctive way of thinking and working which set it apart from other disciplines and subjects, and this thinking is labelled as CT.

Somewhat in the middle of the aforementioned spectrum, Singapore adopts an opt-in basis for schools with regards to teaching students CT; most of these instances are enrichment programmes such as Code for Fun. Code for Fun is an enrichment programme jointly designed by IDA (Infocomm Development Authority) and MOE (Ministry of Education) of Singapore specifically for Singapore primary and secondary schools to increase their exposure to coding and CT ([12]). Outside of school, Singaporean parents have grown to be increasingly aware of the value and importance of CT and coding as they generally see it as an added advantage their children to be successful in their future ([23]). As such, many uncoordinated private enrichment programmes emerged over the recent five years to cater to such an increasing demand, especially when Singapore started pushing for her development as SMART Nation since 2014 ([24]).

For the formal curriculum in Singapore, Computing is offered as an examinable subject in certain schools at both the Ordinary Level and the Advanced Level. However, these courses only impact a small percentage of students (approximately 0.5%, although this number is growing). If schools are commissioned to equip young people for social and civic participation in a world impacted by the rapid growth of computing technologies, it is critical that in the near future all students must acquire a core body of knowledge related to computing.

Limitations and constraints pertaining to resources, professional development, time and school leadership are real. Teaching CT as a separate subject is commonly viewed as costly, and even risky. Many practical problems which plagued the implementation of teaching Computing or CT at schools in the UK were reported comprehensively in [22]. Many teachers expressed their concerns about their own content knowledge of Computer Science and programming, their lack of confidence in directing students in tackling hands-on problems. Furthermore, they were in search of the right pedagogy for teaching Computing to students, and often challenged by

[1] In this context, 'effective' just means computable, i.e., processes implementable by a computer program.

students' wide ranging background experience in computer programming, and had to cope with teaching CT in such a differentiated learning landscape.

An alternative way out of this problematic situation is *not* to teach Computer Science or CT as a separate subject but to broaden CT education by *integrating CT with STEM*[2]*-based subjects*. In particular, it was suggested in [5] that teaching CT provides a highly workable means to bridge the existing STEM skill gaps with regards to STEM education. A natural candidate subject for such an integration is Mathematics. Unsurprisingly, there are many similarities that Mathematical Thinking (MT) and Computational Thinking (CT) share. MT is the paradigm for acquisition of mathematical concepts, skills, and most importantly, problem solving. Already in [10], the parallel between CT (restricted to coding) and MT has been drawn up. Therein it was argued that the core of CT is *Problem Solving*, and in order to empower a computational thinker and problem solver, a framework that comprises (1) Notions, (2) Competencies, (3) Procedures, (4) Disposition, and (5) Metacognition was proposed for a plausible curriculum for CT.

While such the framework proposed in [10] had been actualised in CT-inspired lessons for teachers' professional development at the postgraduate level (see [10, Section 4]), no systematic exploration of this link between CT and MT has ever been carried out in Singapore schools, say at the secondary school level[3]. This paper aims to fill in this gap in the literature; in particular, we want to find answers for the following:

> **Research Question**:
> "Can secondary school mathematics students be taught to think computationally?"

Our ensuing exposition develops along the following lines. In Section 2, we trace back to the origins of the notion of CT to see how CT was first conceived and understood, and how this understanding has changed over time. Next, Section 3 presents a panoramic view of curricular developments in various parts of the world that have attempted to connect CT and MT. This developmental survey adds credibility for the idea of tapping on mathematics as the STEM subject with which CT can be integrated. We then, in Section 4, give details of a pilot study which has begun this year at a certain secondary school in Singapore. Aims and objectives, chosen framework and methodology, the actual classroom implementation, preliminary findings and evaluation of this pilot study will be given here. Finally, Section 5 explicates our future research plans that will take us further along the lines of CT-MT integrated teaching.

2 A peek into history

CT has recently been gaining global attention in the education arena at all levels, from pre-school to tertiary. Arguably, CT has been a key component of how computer scientists and mathematicians 'do their work' since the 1960s, about the same time when modern desktop computers were commercialized. However, CT as a terminology did not appear until Seymour Papert – a mathematician, computer-scientist and educator – came up with this original and powerful idea 'Computational Thinking' in his 1980 book '*Mindstorms*' ([16, p. 182]). The

[2]STEM stands for Science-Technology-Engineering-Mathematics.
[3]Secondary School students are aged from 13 to 16 in Singapore.

details of CT, as envisioned by Papert, were not given in that book but were only later expounded in [17]. Unfortunately, his ideas were too far ahead of his time (and we shall come back to elaborate on that in a moment) and hence did not receive broad attention in education.

The term 'CT' was only recently popularized by Jeannette Wing ([27]) who proposed a new set of definitions for it and gave a cry to action for CT to be learned by *all* school children. Wing labels CT an "attitude and skill set" that everyone can learn and use. The emphasis is on solving problems by exploiting the fundamental concepts of computer science: abstraction, decomposition, recursion, separation of concerns, and so on. In a nutshell, Wing equates CT with thinking like a computer scientist.

Wing's call for CT education sparked a vibrant and ongoing debate about the definition and purpose of operationalizing the term 'CT' for K-16 instruction. Voluminous CT definitions and frameworks emerged, proposed by researchers, industries and educators ([19, 25, 9]) alike. In a recent review of CT literature conducted this year that comprises as total of 125 papers, Kalelioğlu, Gülbahar and Kukul [14] presented the most commonly used words associated to CT as a *wordle*, as shown in Figure 1:

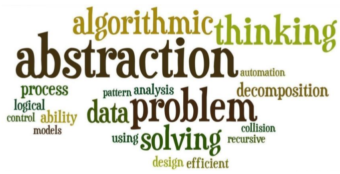

Figure 1: Common words used to define computational thinking

The above set of words do seem to suggest an evidential confluence of opinions by different people of what constitute the key elements of CT, though no consensus has been reached. Worse still, the terminological vagueness is further accentuated with suggestions that CT should be integrated into school subjects other than Computing. Indeed, some scholars even viewed CT as a third way of doing science – the other two being theoretical and empirical approaches (see [7]).

As for mathematics, a similar conclusion was reached by Rambally in [20], where this comprehensive study of the relationship between CT and MT reveals that CT is more than just a form of applied MT. Indeed, the popular definition of CT fashioned by Wing (as abstraction, algorithmic thinking, decomposition, problem solving, and so on) is far too restrictive and suffers from a lack of distinguishing traits that would make CT compelling or worthwhile for core subject teachers to rethink about their current practices and make room for CT in their curriculum.

It is helpful now to revisit Papert's original interpretation of CT. In the introduction to *Mindstorms* ([16]), Papert pointed to the rise of the personal computer, and how the personal computer might permeate everyday life and work. But he was visionary to propose how computers may influence the way people think and learn:

> "A few talked about the computer as a teaching machine. This book poses the question of what will be done with personal computers, but in a different way. I shall be talking about how computers will affect the way people think and learn. I begin to characterize my perspective by noting a distinction between two ways computers might enhance thinking and change access to knowledge."

What Papert envisioned was that programming is a way of talking with the computer and reasons that "learning to communicate with a computer may change the way other learning takes place". In other words, CT has never been about machines but it has *always* to do with the mind, to do with thinking and learning!

Elsewhere in a 2018 conference paper, [15], it was shown that CT framework can be more meaningfully developed from a disciplinary perspective, i.e., how professionals in CT-integrated fields make use of CT and how can these knowledge translate into classroom experiences in schools to "bridge the skills transition from school to work". The point to gather from all the above-mentioned arguments is that one does not begin with a set of generic CT skills or concepts and then tries to connect them to similar and often contrived ones in other disciplines. The right way round should be the focus on how CT is already embodied in discipline-specific forms from a practitioner's standpoint.

3 A link to make

3.1 Domain-specific CT

Our preceding overview of the history of CT and its subsequent evolution, coupled with recent research findings, all point to potentially deeper, more authentic integration of CT and school subjects by examining how CT is used in the subject itself as well as its manifestations in the workplace. We conjecture that defining CT specifically for mathematics learning would (1) address criticisms of vagueness in defining CT, (2) help us better understand the relationship between CT and MT, (3) be more easily understood and accepted by mathematics educators, and consequently, (4) lead to more focused CT research by being very specific and precise about the phenomenon being studied. In this paper, we propose to define CT for integration with mathematics learning that is specific to the domain of secondary school mathematics. This definition will bring us to better focus in answering the question of whether secondary school mathematics mathematics students can think computationally.

What needs to be done at this juncture is a brief survey of historical efforts to integrate CT and MT. Because domain-specific CT aspects are often deeply embedded and intertwined with the educational interventions, making these implicit aspects explicit can be challenging. It is therefore helpful for us to (1) infer and compare historical attempts of CT-MT integration, (2) consider varying practices of CT-MT integration, and (3) evaluate the practicality of some of these intervention designs.

3.2 Historical attempts

Seymour Papert developed the first version of *Logo programming* as a tool for learning mathematics by *constructionism*: through participation in project-based learning, students connect across different ideas and domains of knowledge with suitable scaffolding and facilitation by the teacher instead of step-by-step instruction ([26]). A 1979 study led by Papert and his MIT colleagues ([18]) culminated with a 144-page report which described episodes of students' learning of geometrical concepts, together with a myriad of mental pathways students took to access these geometrical ideas. Notably, Papert and his team intended that Logo function as a vehicle to build and reflect on knowledge. Such an intention set the right conditions for Logo to thrive during the 1980s ([1]). However, Logo subsequently developed into a computer tool mainly for drawing geometrical shapes and thus vastly departed from its intended purpose of revolutionizing children's mathematics learning.

Two modern day examples of CT-MT integration, namely *Computer Based Mathematics* (CBM) and *Code by Math*, took a rather radical approach. CBM positions Mathematics as an anchor for computational thinking across all subjects. Specifically designed curricula aim for students to learn mathematics on a call-by-need basis in the process of building computational models to pursue questions of interest. This is in line with what Papert called "*project before problem*", where projects are the primary entities and mathematical problems appear in the course of the projects' development ([17]). Code by Math goes along the line of Papert's conception of *synergistic relationship* between mathematics and coding: the two are best learnt together as one will reinforce the other. The big idea that Code by Math operates on is that students learn mathematics and coding concurrently within one comprehensive and coherent curriculum so that it becomes second nature to learners that mathematical ideas can be constructed, explored and expressed via coding. By so doing, one would be able to side-step the problem of deciding when and how students should acquire coding in their mathematics learning.

Yes another a milder version of curricular reform for CT-MT learning theorizes that the relationship between CT and MT is one of mutual enhancement, grounding of concepts, and bringing new perspectives in learning. An example of this was the *ScratchMaths* research program that developed a 2 year programme where students aged 9 – 11 learn Scratch programming with focus on computational thinking during the first year, and then mathematical thinking through an integrated programming-mathematics curriculum in the second year. This program is currently piloted in some 50 primary schools in United Kingdom. Benton, Hoyles, Kalas and Noss ([4]) reported that student engagement and deep learning of certain topics were evident under conditions that include teachers' fluency with the programming tool and implementation fidelity.

3.3 What is there for Singapore?

High-stake examinations such as the Primary School Leaving Examination (PSLE), the 'O' and 'A' Level-examinations play an important role in the Singapore education system. Understandably, the Singapore Mathematics Curriculum cannot afford to make too radical a change in its curriculum design as it may affect several school levels. Moreover, teaching CT or Computing as a separate subject would take away the limited resource of time. Hence a more extensive curriculum reform to integrate CT and MT would be unrealistic and, in fact, unwise in practice.

What would have been a more prudent move for Singapore is to take a step towards developing the synergistic relationship between CT and MT. In other words, it is much more viable for teachers to identify specific mathematics topics and their associated aspects of MT that can potentially exploit CT to bring about a deeper understanding and appreciation for both the mathematics topic itself and the computational aspect of it. Furthermore, in order that the CT-MT integration programme to be sustainable, the onus ought to be on the teachers to integrate CT into mathematics teaching and learning – this teacher-initiated integration must take place within the usual classroom parameters and constraints ([11]).

4 A look in the class

Finally we arrive at the *plat principal* of this paper. Let us begin by presenting the theoretical framework, APOS, on which we ground our preliminary research.

4.1 Theoretical framework: APOS

The APOS (Actions-Processes-Objects-Schemas) model is a constructivist theory of how learning mathematics might take place and was developed by Dubinsky and others ([8]), out of an attempt to elucidate Piaget's concept of *reflective abstraction* in learning, with particular focus to mathematical learning. [3] explained that

> "an individual's mathematical knowledge is her or his tendency to respond to perceived mathematical problem situations by reflecting on problems and their solutions in a social context and by constructing or reconstructing mathematical *actions*, *processes*, and *objects* and organizing these in *schemas* to use in dealing with the situations."

Models of specific mental constructs, called *genetic decompositions*, for learning mathematical concepts are developed as working hypotheses that can be tested experimentally. A genetic decomposition comprises two parts at each stage: a description of the specific mental constructs that occurs at that stage, together with what evidence is there to prove that students have already built those constructs. Precisely because genetic decomposition describes the mental constructs needed to learn a new mathematical concept, it can be used to design a lesson that explicitly incorporates CT to build mental constructs at each APOS stage. Every genetic decomposition can be enacted using the Activities-Classroom discussion-Exercise (ACE) pedagogy cycle ([2]).

4.2 Background of pilot study and methodology

The Head of Department of Secondary School A invited the first author to be their curriculum expert. They needed help to create a teaching innovation which incorporates CT into mathematics. School A saw the potentials of collaboration between mathematics researchers and school mathematics teachers with the aim to create a positive impact on the teaching and learning of mathematics using CT-inspired pedagogy. Meetings with a *teaching team* of four members decided that: (i) the team is ultimately responsible for designing, implementing and refining the teaching innovation that exploits CT-MT integration; (ii) the curriculum expert executes the initial kick-off for lesson planning, implementation, and evaluation, and (iii) the curriculum expert provides continual support by offering expert advice to ensure pedagogical enhancement and programme fidelity.

The programme was to be piloted for only one class in the first to second term (i.e., somewhere within first 3 months of the school year 2018), and the identified subtopic was that of Rational and Irrational Numbers, which lies within the larger topic of Real Numbers. This subtopic was chosen by the teaching team because it recognized that the concepts of rationality and irrationality of real numbers were often glossed over by students, and even teachers for the reason that these concepts appeared abstract and not related to other parts of the chapter on real numbers. Furthermore, this problematic situation was not helped by an extremely limited range of test items by which one could assess the students' understanding of the subtopic – there is essentially just one question type which students could easily attempt with little or no understanding of rationality and irrationality (see Figure 2).

Sample question
Of the given numbers below, circle only those which are irrational:

$$-1.01,\ 5\frac{2}{3},\ \sqrt{7},\ 2.3\cdot 1,\ \sqrt[3]{\frac{64}{27}}.$$

Figure 2: Item type: Identification of rational/irrational numbers

More importantly, the teaching team wanted students to have a deeper and tangible concept of rationality/irrationality beyond what was merely tested. However, this is not a easy issue to circumvent because the traditional definition of a rational number is intrinsically 'unhelpful': irrationality phrased in terms of the negation of the traditional definition of rationality is extremely hard to verify.

The team then zoomed into the mental constructs needed to understand the mathematical concepts of rationality and irrationality. Traditionally, one has:

Definition 1 (Rationality (Traditional)) *A* rational *number is one which can be expressed in the form of $\frac{a}{b}$, where a and b are integers, and $b \neq 0$.*

A number which cannot be expressed in the above form is called *irrational.*

The main difficulty about this definition is centred around the issue of *representation.* That $\frac{1}{3}$ is rational follows directly from this definition but reasoning that 0.512 is rational is slightly more involved, and even more so $0.51\dot{2}$. Guided by the team's past teaching experience, it was agreed that showing the first example of an irrational number always proved challenging. The common first examples are $\sqrt{2}$ and π, which are very unconvincing. According to this traditional definition of irrationality, one has to check from an infinitude of integers a and non-zero integers b that r is not equal to $\frac{a}{b}$ – an impossible feat! One simply does not get a handle on reasoning with irrational numbers.

Applying APOS theory, we can now articulate such difficulties:

- Action: Test whether a given number r is rational means to check whether r satisfied the condition specified in Definition 1. This action identifies the positions of the 'candidate' numerator and denominator that can represent the number r.

- Process: Look for possible pairs integers a and b (b must be non-zero) amongst an infinite of such possibilities satisfying the condition that $r = \frac{a}{b}$.

- Object: The object of a rational number is the static entity comprising a pair of integers a and b (where $b \neq 0$), together with the linking arithmetic operation of division, since $\frac{a}{b}$ can be interpreted as the value obtained when a is divided by b. For instance, this object image can be in the form of a 'pizza-representation' of a fraction.

- Schema: A (seemingly impossible) procedure of verifying rationality relying on integral division, performing this operation over a possible infinite set of integers.

To circumvent this intrinsic difficulty of representation, the curriculum expert then proposed the CT concept of *data representation*. At Secondary One, students had already learnt about decimal representations of real numbers. This gives the teaching team the idea of exploiting:

Theorem 2 $x \in \mathbb{Q} \iff x$ *has either a terminating or recurring decimal representation.*

Note that aspects of CT to mathematics were not 'force-matched'. Instead, by looking deeper into the learning difficulties encountered in specified mathematical concepts and exploiting CT appropriately, the team created intentional opportunities for better and deeper understanding of the concept in question. As promised in Subsection 3.1, we made explicit the CT concept of rational numbers through *data representation using decimals*.

4.3 Implementation

We can now reap the fruits of our labour. Instead of the traditional definition of rational numbers, the team proposed the following CT-inspired definition:

Definition 3 (Rational number (CT-inspired)) *A* rational number *is one which has either a terminating or recurring decimal representation.*

Given Definition 3, the mathematical knowledge of rationality of a number can now be explicated within the APOS framework as something concrete and useful:

- Action: Test whether a given number r is rational means to check whether r satisfied the condition specified in Definition 3. This action identifies with the decimal representation of the number r, which is concrete.
- Process: Examines the pattern nature in the decimal representation of r, and decide if it is terminating, recurring or, perhaps, neither of these.
- Object: The object of a rational number is the static entity of its terminating or recurring decimal representation. The image is a string of digits which is concrete and can be easily manipulated syntactically.
- Schema: A procedure of verifying rationality visually by inspecting the decimal representation of the number concerned.

Based on the identified APOS characterization of the concept of a rational number. The team laid down the following scheme of work for teaching the sub-topic of Rational/Irrational Numbers that translates into a suite of three CT-MT integrated lessons (L1, L2 and L3):

L1. The main objective of (L1) is for the students to establish, through the ACE-cycle, Theorem 2, i.e., the equivalence of the CT-inspired definition of the rational number to its traditional definition. The pairing of CT-MT concepts essential for the lesson development of (L1) are:

  - CT: Algorithm; MT: Division algorithm
  - CT: Data representation; MT: Decimal representation

- CT: Data transformation; MT: Fraction-to-decimal conversion; and decimal-to-fraction conversion
- CT: Finite nature of discrete systems; MT: Pigeonhole principle

L2. The second lesson focuses on the position of Irrational numbers in the classification chart (Figure 3).

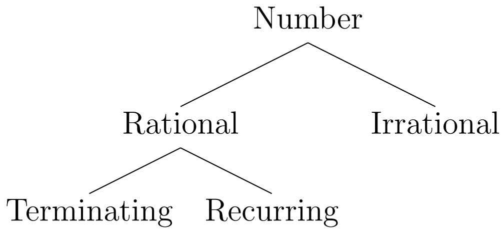

Figure 3: Classification chart for real numbers

Based on the CT-inspired definition of Rational Numbers, the lesson aims at scaffolding the students to build their own irrational numbers as ‘first examples’ of irrational numbers. The salient pairing of CT-MT concepts essential for the lesson development of (L2) are:

- CT: Data representation; MT: Decimal representation
- CT: Pattern recognition; MT: Recurring/Non-recurring patterns

L3. The third lesson introduces simple recursion to produce the decimal approximations of $\sqrt{2}$, and help students appreciate the infinite character of real numbers such as $\sqrt{2}$. Here, ‘just-in-time’ teaching of Excel commands for recursion within the spreadsheet environment is to take place.

Due to the limitations of the length of exposition, we use (L2) to illustrate how the ACE-cycle was set into action. The instructional objective of (L2) is for students to produce their own ‘first examples’ of irrational numbers. Students were tasked with the **Activity** to convert $\frac{1}{99}$ to decimal form as taught in (L1), and decide if it is rational. This activity consolidated the skills of long division, the notation for representing recurring decimals and the characterization of rational numbers in the sense of Theorem 2. During the **Classroom discussion** that followed, the instructor engaged the class in discussion about creating an irrational number, with special focus on the defining qualities of an irrational number. This discussion prepared for the **Exercise** of constructing an irrational number based on ‘Think-Pair-Share’. Individual students began by thinking about a possible decimal representation of an irrational number. Then in pairs, students would discuss and finalized on the formulation of their first irrational numbers. Finally, selected pairs of students presented to the rest of the class their irrational numbers, and explained the CT-inspired strategy they exploited to create these irrational numbers. To help the students, it was suggested by the expert to form the required decimal representation using only 0’s and 1’s.

4.4 Notes and preliminary findings

4.4.1 Classroom notes

The lesson suite (L1–L3) was taught to a class of 30 Secondary One students, aged 13, with mixed abilities. Each lesson took about 50 minutes (i.e., one period in the school timetable). The lesson was taught by the curriculum expert in the initial stages of the teaching innovation – this was understandable since the teachers needed to see for themselves how a CT-MT integrated lesson actually runs. Teacher S a member of the teaching team acted as co-teacher who was in charge of addressing any learning difficulties faced by students.

During the *Activity* part of (L2), the majority of the students could recall from (L1) the fraction-to-decimal conversion procedure. For instance, they could, from the recurring nature of $\frac{1}{99}$, deduce that $\frac{1}{99}$ is rational. A few students also mentioned the above answer can be arrived at directly from the traditional definition of rationality. The instructor then took this remark further during the *Classroom discussion* by 'toying' with the idea of constructing an irrational number. The technical word 'constructing' was explained to mean 'building' or 'making out'. The students then appreciated that they had to freedom to form their own irrational numbers – they quickly realized that the whole class might come up with many different irrational numbers.

The statement of the *Exercise* of formulating their 'first examples' of irrational numbers was given, and the students were asked to carry out the Think-Pair-Share, an exercise mode which they were familiar. The following are some irrational numbers they contributed:

I. $0.101001000100001\ldots$ II. $0.010110111011110\ldots$ III. $0.10110011100011110000\ldots$

Of the 15 pairs, at least 6 pairs came up with the response (I). This coincidence created such an impact on the students that for a second, some thought that the answer was unique. This 'misguided' thought was quickly dismissed by two other pairs who boldly proposed alternative responses (II) and (III).

Each pair was given some time to talk about the numbers they created and why they thought these numbers were irrational. During their presentations, students described how they came up with the decimal presentations that would 'defy' the CT-inspired definition of a rational number. Notably, for the first pair who came forward to share their answer, the student in charge of writing down the answer turned to the instructor and said that he could go on forever and asked if he could stop the writing up to some decimal places. He then put a few trailing dots after the last digit he wrote. The other student explained the rule that produced the pattern in (I) which ensured that the decimal representation was non-recurring, and he went on to explain that the pattern was to go on indefinitely (pointing to the trailing dots) and hence the decimal representation must be non-terminating.

Each pair showed great enthusiasm in presenting the irrational numbers which they created. There was very little need on the part of the instructor to prompt the students in their explanation as they took ownership of their own inventions. To heighten the learning atmosphere, the instructor named each of the irrational constants after their creators – a practice, which the instructor added is a common practice within the mathematics community.

4.4.2 Post-lesson discussions

Following the lessons, post-lesson meetings were conducted after the teaching team had watched all the three lesson video recordings. The general opinion was that the students were engaged in learning the mathematical concepts of rational and irrational numbers. Pertaining to (L2), the teaching team was heartened by the students' engagement with the challenging exercise at hand as well as their enthusiasm shown during their pair-presentation. Additionally, the teaching team realized for themselves that a CT-MT integration lesson need not always involve coding or programming of any kind. There were also some concerns raised by the team: (1) The CT-inspired definition was not the 'official' definition, and there may be a danger that students would write this as the definition for rationality of a number instead of the traditional one. (2) The topic of constructing irrational numbers is not within the Secondary School Mathematics Syllabus. It was not clear how much the students appreciated what was going on in the lesson; for instance, did they know they have done something mathematically non-trivial and meaningful? (3) The CT-MT approach was interesting but teachers might not be ready to design such a lesson, let alone conduct one. In the long run, for CT-MT integration to be sustainable there will be a need for more mathematics teachers to receive professional development and training in this kind of mathematics pedagogy. (4) It was not clear how the CT-inspired definition of rationality can be used to prove that certain constants such as $\sqrt{2}$ is irrational.

4.4.3 Some remarks and preliminary findings

The lesson observation notes confirmed that the lesson taught embodied the genetic decomposition for the concept of rational/irrational numbers (see the APOS framework applied in Section 4.3). The lesson enabled students to use CT-inspired reasoning methods to construct a schema for determining whether a number is rational or irrational based on its decimal form showing non-termination and no-recurrence. By so doing, the students understood several deeper mathematical peculiarities of irrational numbers, such as (a) there are infinitely many irrational numbers, (b) irrational numbers can possess regular patterns and do not seem so exoteric or strange-looking such as the usual irrational constants, $\sqrt{2}$ and π, found in their textbooks, and (c) any irrational number can be approached by a sequence of rational numbers with increasing accuracy.

In the process of explaining why the numbers the student-pairs created were indeed irrational, the students produced their own rules for generating the required pattern. The production of such rules provides concrete evidence that CT was driving their cognitive process – they were assuming the role of a human providing instructions to a computer to produce an output by applying a precise set of instructions. For example, the first pair of students who presented the response (I) explicitly stated the rule they used to generate the pattern:

> "First, we print one '1', followed by one '0', and then another '1', and followed by two 0's. Each '1' is separated by an increasing number of 0's. We increase the number of 0's by one each time. So the 1's got spaced out further and further ..."

All in all, what these students did – the arguments used, the rules applied and the deeper insight obtained of rational numbers – gave us confidence in claiming that *secondary school mathematics students can be taught to thinking computationally*!

It appeared that this particular mathematics exercise or problem belongs to a category of problems identified by G. Rambally in [19] that are easier to formulate and more efficiently solved using CT.

The lesson planning and design were guided by APOS theory. It is worth noting that CT only came into the scene after the relevant issues concerning MT have been raised and discussed. The teaching team applied this approach as a basis of creating the opportunity for CT to be relevant to mathematics, and not the other way round. This approach was faithful to our initial stance that for CT to be relevant to mathematics learning we must look for aspects of CT which are domain-specific. In this case, an accurate genetic decomposition of rationality/irrationality of numbers effectively surfaced those pedagogical problems that plagued the traditional teaching methods. This allowed the curriculum expert exploit CT to address the targeted problem at hand.

5 A direction to take

The APOS framework appeared to be handy in identifying the areas in mathematics teaching and learning that can be addressed or enhanced by integrating CT with MT. The implemented suite of three lessons (L1 – L3) appears promising in opening up a new avenue for developing and implementing CT-MT integrated mathematics lessons relevant to the local Singapore contexts and sensitive to the constraints of an everyday mathematics classroom.

However, there are obvious limitations of our findings and conclusions drawn from this pilot study. Firstly, the evidence base is far too limited and the time-frame too short: our remarks were based merely on the observation notes from one lesson for a particular class. Indeed systematic analyses of the lesson observation notes and the post-lesson discussion are lacking. We could hardly predict the effectiveness of this CT-MT approach for different sub-topics, students, teachers and time-frames. At the moment, the teaching team has not decided the most appropriate mode of assessment to measure the outcomes of this teaching innovation. Indeed, we are not even clear whether it is the CT or the MT (or both) that we should measure.

School A is currently entering into the next phase of the collaborative programme, which is to review what has transpired for the past two terms and what will be done for the next two terms. The second phase will involve the teaching team in applying the APOS framework to design and implement a second suite of CT-MT integrated lessons on another topic. In this phase, the curriculum expert will switch to a more supportive role of advising the team about lesson design and implementation issues.

One possible direction for our future work is to design a more fine-grained analysis of the relationship between CT and MT during the implementation of CT-MT integrated lessons. Such a research enterprise will produce a set of design principles for CT-MT integrated activities and build a local evidence base to determine whether and how CT can enhance mathematics learning and vice versa.

References

[1] Agalianos, A., Noss, R. and Whitty, G. (2010). Logo in Mainstream Schools: the struggle over the soul of an educational innovation, *British Journal of Sociology*, **5692**(786636649).

Available at https://doi.org/10.1090/0142569012009444.

[2] Arnon, I., Cottrill, J., Dubinsky, E., Okta c, A., Roa Fuentes, S., Trigueros, M., and Weller, K. (2014). *APOS Theory*. Availabe at https://doi.org/10.1007/978-1-4614-7966-6.

[3] Asiala, M., Brown, A., DeVries, D. J., Dubinsky, E., and Matthews, D. (1997). *A Framework for Research and Curriculum Development in Undergraduate Mathematics Education* (Vol. 40).

[4] Benton, L., Hoyles, C., Kalas, I., and Noss, R. (2017). Bridging Primary Programming and Mathematics: Some Findings of Design Research in England. *Digital Experiences in Mathematics Education*, 115 – 138.

[5] Cortez, M. B. (2016). Teaching Computational Thinking is the First Step to Bridging STEM Skills Gap. EdTech November issue. Available at https://edtechmagazine.com/k12/article/2016/11/teaching-computational-thinking-first-step-bridging-stem-skills-gap.

[6] Cuny, J., Snyder, L. and Wing, J. M. (2010). Demystifying computational thinking for non-computer scientists. Unpublished manuscript in progress, referenced in http://www.cs.cmu.edu/~CompThink/resources/TheLinkWing.pdf(2010). Key: citeulike:13256108.

[7] Denning, P. J. (2017). Computational Thinking in Science. Retrieved from https://www.americanscientist.org/article/computational-thinking-in-science.

[8] Dubinsky, E., and McDonald, M. A. (2001). APOS: A constructivist theory of learning in undergraduate mathematics education research. *The Teaching and Leaning of Mathematics at University Level*, 275–282.

[9] Google. (2018). Computational Thinking for Educators. Retrieved from https://computationalthinkingcourse.withgoogle.com/course?use_last_location=true.

[10] Ho, W. K. and Ang, K. C. (2015). Developing Computational Thinking Through Coding. In Yang, W. C., Meade, D., and Liu, C. (Eds.) Proceedings of the Twentieth Asian Technology Conference in Mathematics, Leshan, China: Mathematics and Technology, LCC, 73 – 87.

[11] Ho, W. K., Yap, R. A. S., Tay, E. G., Leong, Y. H., Toh, T. L., Quek, K. S., Toh, P. C., and Jaguthsing, D. (2018). Understanding the Sustainability of a Teaching Innovation for Problem Solving: A Systems Approach. In Liljedahl, P. (Ed.), *Mathematical problem solving: Current themes, trends and research* (pp. 1 – 19). Burnaby: Springer.

[12] IDA. (2018). Official website of Code for Fun by Infocomm Development Authority of Singapore. Available at http://www.ducklearning.com/code-for-fun-enrichment-program-by-ida/.

[13] Jones, S. P., Mitchell, B. and Humphreys, S. (2016). Computing At School in the UK. Report under review by CACM, referenced in https://www.microsoft.com/en-us/research/wp-content/uploads/2016/07/ComputingAtSchoolCACM.pdf.

[14] Kalelioğlu, F., Gülbahar, Y., and Kukkul, V. (2016). A Framework for Computational Thinking Based on a Systematic Research Review. *Baltic J. Modern Computing*, **4**(3), 583 – 596.

[15] Malyn-smith, J., Lee, I. A., Martin, F., Grover, S., Evans, M., and Pillai, S. (2018). Developing a Framework for Computational Thinking from a Displinary Perspective. In *Computational Thinking on Education 2018*, 182–184, Hong Kong.

[16] Papert, S. (1980). *Mindstorms: children, computers, and powerful ideas*, Basic Books, Inc. New York.

[17] Papert, S. (1996). An Exploration in the Space of Mathematics Educations, *International Journal of Computers for Mathematical Learning*, **1**(1), 95 – 123.

[18] Papert, S., Watt, D., DiSessa, A., and Weir, S. (1979). *Final report of the Brooline Logo Project: Part II: Project summary and data analysis*, Logo Memo 53.

[19] Rambally, G. (2014). Integrating Computational Thinking in the Mathematics Curriculum. In M. Searson and M. Ochoa (Eds.), Proceedings of SITE 2014–Society for Information Technology and Teacher Education International Conference, 120 – 127. Jacksonville, Florida, United States: Association for the Advancement of Computing in Education (AACE). Retrieved July 12, 2018 from `https://www.learntechlib.org/primary/p/130722/`.

[20] Rambally, G. (2016). The Synergism of Mathematical Thinking and Computational Thinking. In *Cases on Technology Integration in Mathematics Education*, 416 – 437. Retrieved from `https://doi.org.10.4018/978-1-4666-6479-5.ch021`.

[21] Robertson, A. and Olson, S. (2013). *Sensing and Shaping Emerging Conflicts: Report of a Workshop by The National Academy of Engineering and United States Institute of Peace*, The National Academies Press, Washington D. C.

[22] Sentance, S. and Csizmadia, A. (2017). "Computing in the curriculum: Challenges and strategies from a teachers perspective", *Educ Inf Technol*, **22**, 469 – 495.

[23] Hio, L. (2016). Parents see programming, computing skills as essential for future in a Smart Nation (2016, March). The Straits Times. Retrieved from `https://www.straitstimes.com/tech/coding-classes-for-kids-in-high-demand`.

[24] Teng, A. (2014). More kids to learn programming in Smart Nation push (2014, December). The Straits Times. Retrieved from `http://origin-stcommunities.straitstimes.com/education/primary/news/more-kids-learn-programming-smart-nation-push`.

[25] Weintrop, D., Beheshti, E., Horn, M., Orton, K., Jona, K., Trouille, L., and Wilensky, U. (2016). Defining Computational Thinking for Mathematics and Sciences Classrooms. *Journal of Science Education and Technology*, **25**(1), 127–147.

[26] Wikipedia. (2018). Constructionism (Learning Theory). Retrieved from Wikipedia at `https://en.wikipedia.org/wiki/Constructionism_(learning_theory)`.

[27] Wing, J. M. (2016). Computational Thinking. *Communications of the ACM*, **49**(3), 35–35.