

Real-world problems through computational thinking tools and concepts: the case of coronavirus disease (COVID-19)

Hatice Beyza Sezer and Immaculate Kizito Namukasa
*Department of Curriculum Studies, Western University Faculty of Education,
London, Canada*

Abstract

Purpose – Many mathematical models have been shared to communicate about the COVID-19 outbreak; however, they require advanced mathematical skills. The main purpose of this study is to investigate in which way computational thinking (CT) tools and concepts are helpful to better understand the outbreak, and how the context of disease could be used as a real-world context to promote elementary and middle-grade students' mathematical and computational knowledge and skills.

Design/methodology/approach – In this study, the authors used a qualitative research design, specifically content analysis, and analyzed two simulations of basic SIR models designed in a Scratch. The authors examine the extent to which they help with the understanding of the parameters, rates and the effect of variations in control measures in the mathematical models.

Findings – This paper investigated the four dimensions of sample simulations: initialization, movements, transmission, recovery process and their connections to school mathematical and computational concepts.

Research limitations/implications – A major limitation is that this study took place during the pandemic and the authors could not collect empirical data.

Practical implications – Teaching mathematical modeling and computer programming is enhanced by elaborating in a specific context. This may serve as a springboard for encouraging students to engage in real-world problems and to promote using their knowledge and skills in making well-informed decisions in future crises.

Originality/value – This research not only sheds light on the way of helping students respond to the challenges of the outbreak but also explores the opportunities it offers to motivate students by showing the value and relevance of CT and mathematics (Albrecht and Karabenick, 2018).

Keywords CT, Programming languages, SIR model, COVID-19

Paper type Research paper

© Hatice Beyza Sezer and Immaculate Kizito Namukasa. Published in *Journal of Research in Innovative Teaching & Learning*. Published by Emerald Publishing Limited. This article is published under the Creative Commons Attribution (CC BY 4.0) licence. Anyone may reproduce, distribute, translate and create derivative works of this article (for both commercial and non-commercial purposes), subject to full attribution to the original publication and authors. The full terms of this licence may be seen at <http://creativecommons.org/licenses/by/4.0/legalcode>

Funding: This work was funded by the Teaching Fellowship of Dr. Namukasa (inamukas@uwo.ca) on Maker Education at the Center of Teaching and Learning, Western University.

Availability of data and material: Links to the website apps which are the sources of our data are provided.

Code availability: Links to the code for the simulations shown in this manuscript are provided in the main text.

Authors' contributions: All authors contributed equally to this work.

Conflicts of interest/Competing interests: The authors have no conflict of interest.



Introduction

When the novel coronavirus was first identified, countries and the World Health Organization (WHO) could not largely understand the risk and rate at which this disease would culminate into a global crisis. The following questions needed to be answered rapidly by experts when responding to the outbreak: at what rate was the infection going to spread in different populations? How were experts, health officials and policymakers going to effectively convey information that would help in understanding the nature of the outbreak? How were they to demonstrate how the different recommended collective control protocols would alter the spread of the outbreak? As [Shepherd \(2020\)](#) indicates, it became very crucial to make the public comprehend the severity of the health risks of this novel virus and the need to critically interpret and implement the precautions recommended by the public authorities.

Mathematics, along with other disciplines, is essential in helping to understand several aspects of an outbreak. For instance, health officials and policymakers may communicate about the rates, trends and parameters of the outbreak. [Kucharski \(2020\)](#) mentions that mathematics can also help with determining what needs to be done to help control the rates of morbidity and mortality. By providing tools for assessment, analysis and predictions, mathematical modeling has been very vital in efforts by experts from a variety of fields who have investigated the dynamics of both emerging and reemerging infectious diseases and insights drawn from them help policymakers to determine and debate courses of action that may prevent high mortality rates ([Siettos and Russo, 2013](#); [Wang et al., 2020](#); [Yates, 2020](#)).

Many models are about the risks associated with a pandemic, the probability and rate of spread in a population and the effects of possible interventions to reduce morbidity and mortality ([Rodrigues, 2016](#); [Walters et al., 2018](#)). Those mathematical models, however, use sophisticated mathematics that is normally only understood by experts in fields that make or use mathematics. Complex mathematics equations, data displays and graphs make it difficult to comprehend the data and the dynamics behind these mathematics artefacts for nonexperts. Computational and programming tools used to compute the equations or illustrate the visualizations are equally complex, as they require understanding of the tools and languages used; however, interactive illustrations of the mathematical models of outbreak, which utilized recent and less sophisticated tools for computational programming such as block-based and text-based programming languages, make it easy for the mathematical models of outbreak to be read and understood by the public ([Froese, 2020](#); [Resnick, 2020](#); [Yeghikyan, 2020](#)). Many of these models are designed to afford opportunities to experiment with different scenarios, and users may view, study and modify the code for the simulations. Recent school activities also show that students might be able to better understand real-world problems by using programming languages. The principal of the Oklahoma School of Science and Math, Dr. Frank Wang, for instance, claimed that offering students chances to play with the tools including the mathematical models, similar to those that real epidemiologists were using, enabled students to acquire a better idea of the pandemic on their own ([Skarky, 2020](#)).

In this research paper, we focus on computational simulations of outbreak based on susceptible – infectious – recovered (SIR) model which commonly have been used to illustrate the spread of the COVID-19 disease ([Ciarochi, 2020](#)). We selected and analyzed two sample simulations of the SIR model designed using a block-based programming language, Scratch. These simulations, in addition to being dynamic and interactive, have visible code hence they are modifiable, which helps students who wish to experiment with them. We specifically investigated: (1) the ways in which Scratch simulations were accessible by students to comprehend the dynamics of the outbreaks and the response needed to slow down the rate of the outbreak and (2) the extent to which these simple simulations illustrate the impacts of variations in precautions and policies implemented during pandemic crisis, including social/ physical distancing, reduced mobility (through staying at home, isolation or quarantine)

and regular hand washing. Generally, this study aims to shed light on opportunities of using CT tools during the current global health crisis.

Literature review

Computational thinking and its integration with real-world problems

The term CT was coined by Papert (1980) and popularized by Wing (2014) 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 [it] out” (Wing, 2017, p. 8). Hoyles *et al.* (2002) and Wilkerson-Jerde (2014) designed environments of computational programming coupled with computational modeling of mathematics and science concepts, processes and systems, for example, population dynamics (e.g. Stroup and Wilensky, 2014; Wilkerson-Jerde *et al.*, 2015a, b). While focusing on CT, these researchers engaged students in the practices of mathematicians (Wilkerson-Jerde, 2014) and of STEM professionals (Wilkerson-Jerde *et al.*, 2018).

Haduong (2019) maintains that the increasing relationship between power and technology in today’s world makes it even more essential to use digital tools to examine children’s and youth’s experiences, goals and expectations which influence both their current and future lives. To Skovsmose (1994) the role mathematics plays in technological development shows its formatting power on society and shows the potential role of mathematics in helping to positively shape the society. The reality is, however, given the way mathematics is currently taught and the nature of the mathematics curriculum, many students are unable to see the applications of their mathematical knowledge and skills to tasks embedded in real-world contexts. This is because real-world mathematics looks messy and complex than school mathematics and is often hidden or invisible (Taut, 2014).

Furthermore, Lee (2012) asserts that mathematical expressions are not always broad enough to address all the issues encountered in real-life. Lee is among the researchers who argue that integrating computational thinking in mathematics teaching offers help to explore and gain a deeper understanding of certain real-world challenges. Wilkerson-Jerde and Fenwick (2016), for example, observe that by focusing on processes, computational thinking concepts facilitate using CT tools and algorithms to make data manipulation, simulation and analysis easier and more manageable. When students use CT concepts in mathematics as a tool to learn with, they are afforded greater student agency, engagement and access to mathematics concepts that would otherwise be more advanced for their grade levels (Gadanidis, 2017; Namukasa *et al.*, 2017; Sanford and Naidu, 2016). Further, Wilkerson-Jerde *et al.* (2015a, b) identified the complementary role of representations (such as drawings) that emphasize components and relationships with representations (such as animations and simulations) that emphasize change across space and time.

In school settings, coding and modeling are some of the ways in which CT is used to model and visualize real world problems to students. For instance, Sanford (2013) and Sanford and Naidu (2017) indicated that learners may be taught to make use of CT concepts and tools to make simulations and to model solutions to mathematics problems. Many countries are integrating learning expectations on mathematical modeling and coding in their new curricular. In Ontario, new mathematics curriculum for example, it is asserted that the models and simulations may use algebraic or probabilistic reasoning in analyzing, representing and modeling data and provide different levels of experiences that are aligned across different grade levels (OME, 2020). When CT tools are used to help understand and represent a disease outbreak, for instance, the emphasis is on conveying the patterns, relations and processes of the outbreak, students may be guided to estimate the dynamics of the pandemic, to investigate the effects of specific control measures and to analyze and synthesize the available real data.

Context

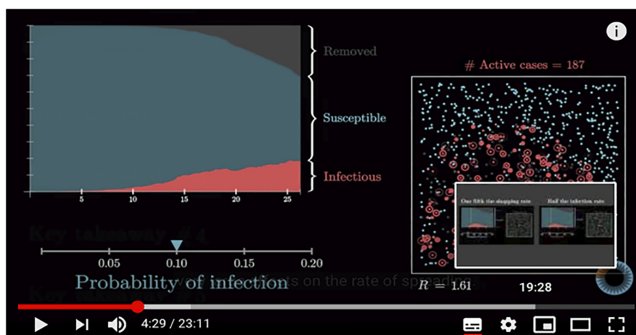
The current contexts of computational tools for understanding outbreak models

With the outbreak, several computational artefacts (e.g. graphical illustration, simulations and data maps) have been shared in the news and on the Internet, particularly on social media. (Adam, 2020; Tahiralli and Ho, 2020; Yates, 2020). Many of these attempted to illustrate to the public audience the spread of the disease (Ciarochi, 2020; Stevens, 2020) and to explain what needed to be done to slow the spread. Examples of these artefacts of the pandemic include: a simulation published by Stevens (2020) and translated in 12 languages; a video illustration of exponential growth by Sanderson [3blue1brown] (2020), in which he explains the probability of infection (transmission), R , and the effectiveness of responses that lower this rate and a graphical illustration on flattening the curve of the spread of the disease [1], which when tweeted by former President Barack Obama was retweeted over 180, 000 times (Barclay and Scott, 2020; Obama, 2020). Whereas certain simulations and graphs only visually illustrate the dynamics of the disease spread, its containment measures and the different possible scenarios, which are dependent on how countries responded, a few others, such as the one by Simler (2020) are interactive simulations that offer manipulable and modifiable interfaces (Namukasa *et al.*, 2016). Simler (2020) explains that the model he offered included:

playable simulations of a disease outbreak. Playable means you'll get to tweak parameters (like transmission and mortality rates) and watch how the epidemic unfolds.

By the end of this article, I hope you'll have a better understanding – perhaps better *intuition* – for what it takes to contain this thing. But first!... [He cautioned] This is not an attempt to model COVID-19. What follows is a simplified model of a disease process. The goal is to learn how epidemics unfold in general. (see Figures 1 and 2).

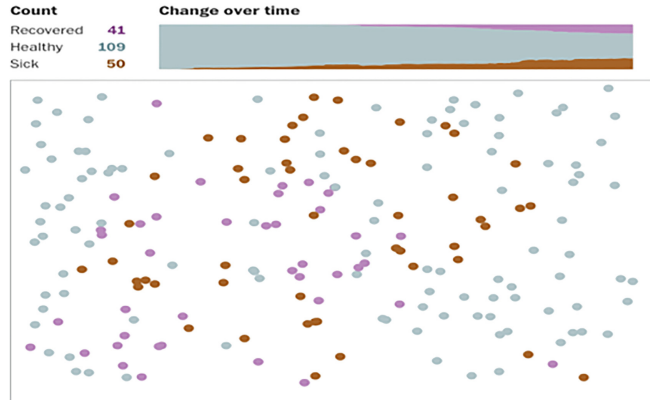
In addition to stand-alone models shared in public media, certain models have been offered by other institutions to provide opportunities for students to improve their comprehension of the pandemic and hence to lead to their informed participation when taking precautions during the pandemic. Further, certain computational thinking communities and initiatives have offered opportunities (e.g. webinars, courses, classes and projects) to promote understanding of the facts and predictions of the disease for students and the public. One example is a course offered at MIT that claimed to apply data science, artificial intelligence, mathematical models and a programming language, Julia, which was repurposed to study the COVID-19 pandemic in 2020 spring semester. According to Raj Movva, a sophomore who took the course, the



Source(s): <https://www.youtube.com/watch?v=Kas0tIxDvrg> & feature =youtu.be

Figure 1.
Exponential growth
and epidemics
(Sanderson, 2020)

To simulate more social distancing, instead of allowing a quarter of the population to move, we will see what happens when we let just one of every eight people move.



Source(s): <https://www.washingtonpost.com/graphics/2020/world/corona-simulator/>

Figure 2.
Simulation of social distancing
(Stevens, 2020)

course provided opportunities for using computation to better comprehend the pandemic and helped in identifying the misinformation about the coronavirus (cited in Miller, 2020).

Another set of examples are simulations shared in online forums such as the National Council of Teachers of Mathematics (NCTM) website and the Scratch online community [2], which is a “vibrant online community of people sharing, discussing, and remixing one another’s projects” (Resnick *et al.*, 2009, p. 60). For instance, the NCTM association published a mathematics simulation [3] on the spread of COVID-19 through social contact to be used as part of the activities for grades 5–12 (NCTM, 2020). The NCTM simulation is interactive and modifiable and shows a dynamic simulation, a graph of cases overtime, four parameters in the parameter pane and a pane for data. Learners are able to ask “what happens if” questions and carry out experiments to answer the questions they are wondering about using the NCTM simulation (see Figure 3).

Method

In this study, we use qualitative research design, specifically content analysis to analyze two interactive simulations, which are selected from the publicly shared online simulations of the current global health crisis. The simulations are designed using Scratch that students might have learned in schools and are modifiable to experiment with different scenarios of reality. We specifically examine the extent to which these simplified, basic mathematical models the Susceptible – Infectious – Recovered (SIR) models are potentially helpful (or not helpful) to make key mathematical and computational concepts of these models understandable by students.

The design of a basic SIR model and its extensions

Epidemiologists use a family of mathematical techniques, called compartmental models, to model infectious disease. In a compartmental model, people are classified into separate groups of people who share the same characteristics and mathematical equations are used to model the processes that affect the movement of people from one classification to another. The SIR model is one of the basic compartmental models, in which the population is split into

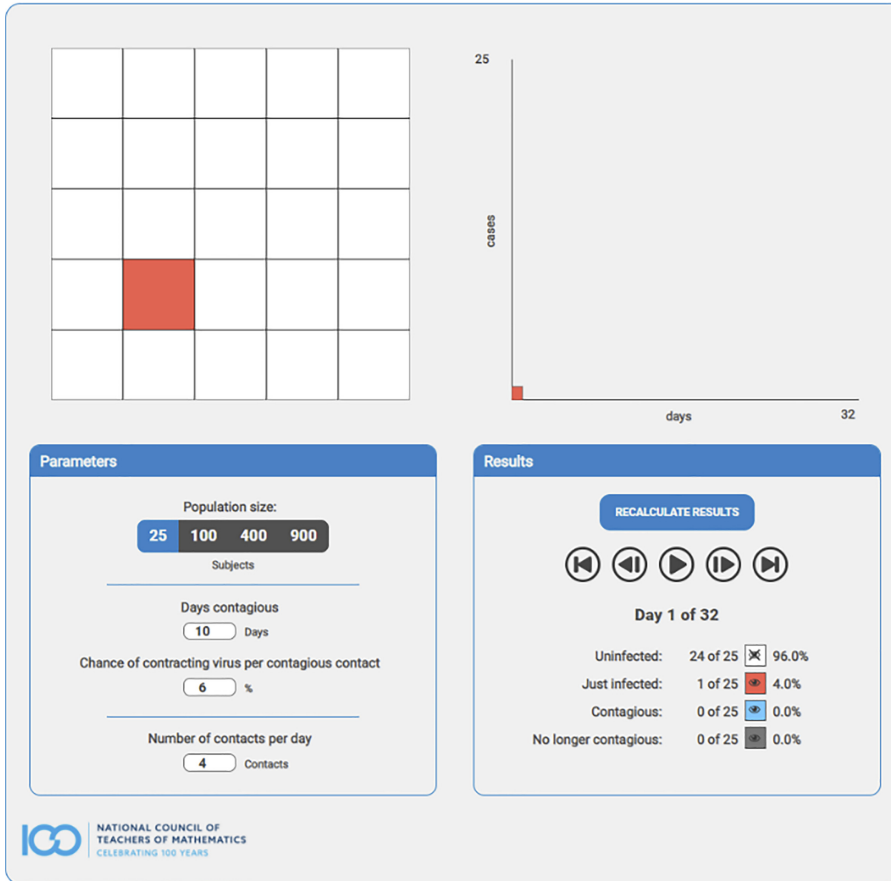


Figure 3.
Simulation of the
impact of social
distancing
(NTCM, 2020)

Source(s): <https://www.nctm.org/tools/pandemic2020/index.html>

three types of people: the susceptible type individuals, S , are the ones who are not currently infected but could get infected; the infected type individuals, I , are the ones who have the disease and henceforth can transmit it to the susceptible and the removed (recovered, immune or dead) individuals, R , are the ones who cannot get infected and cannot transmit the disease to others (Capitanelli, 2020). Two processes are simultaneously at work in this model: first, as a result of contact with an infected individual, a susceptible individual may get infected and move to the infective type. In the SIR model, the number of these movements is proportional to the number of infected and healthy people. Hence the change in susceptible population type is given by the following differential equation:

$$S' = -\beta I \frac{S}{N}$$

where the parameter beta, β , stands for the rate at which infected individuals contact and infect other people, and S' stands for change in S .

The second simultaneous process is that the infected people can enter the removed class, and the number of these movements is proportional to the number of infected people. Hence, we have the following differential equations for the changes in infected and removed:

$$I' = \beta I \frac{S}{N} - \gamma I,$$

and

$$R' = \gamma I,$$

where the parameter gamma, γ , is the rate at which people who are infected people either recover or die. β/γ is R_0 , the total number of people an infected person infects.

Adam (2020) mentions that by grouping individuals into compartments and using mathematical equations to model the interactions between the compartments, the compartmental models do not require an understanding of complex computations. These models, however, are criticized, as they make several assumptions that have not necessarily been observed in real pandemics (Daughton *et al.*, 2017), and the subtleties of many pandemics cannot be comprehensively captured in the simplicity of the SIR models (Yates, 2020). Even though SIR models seem simple, they appear to have been very useful in explaining the needed precautions and responses to the global pandemic crisis (Rodrigues, 2016). Moreover, simple SIR models could be extended to more sophisticated models with four or more compartments, such as by introducing an exposed compartment between S and I or by introducing two simultaneous compartments, recovered and fatal, after I instead of R and several scientists are currently working on these complex models to come closer to the reality of pandemic (Froese, 2020) For instance, Giordano *et al.*'s (2020) model consists of eight compartments for modeling the COVID-19 pandemic in Italy: susceptible (S), infected (I), diagnosed (D), ailing (A), recognized (R), threatened (T), healed (H) and extinct (E), collectively termed SIDARTHE. Their model offers a detailed classification of infected individuals depending on how severe their symptoms are. Extensions of SIR-type models may be further extended to more closely model reality by incorporating demographics and taking diffusion and migration effects along with possible genetic mutations into consideration (Siettos and Russo, 2013).

Results of the content analysis of the simulations of SIR model

We analyzed two different simulations to ascertain the ways in which they are helpful (or not helpful) in illustrating, understanding and responding during the current global health crisis. Through content analysis, we began by examining the dimensions of the simulations and their connections to computational thinking and mathematical concepts. We coded the dimensions found in the selected simulations based on the steps of these simulations. Following coding, we interpreted the presence of the mathematical and computational concepts extracted from the content analysis.

The results of the analysis show that the simulations have the following common dimensions: (1) initialization (the creation of people, their initial location and their initial condition in terms of infection), (2) movements (the direction of people's movement, which is reduced if they choose not to stay at home), (3) transmission of infection (the probability of getting infected when in contact with an infected person, which may depend on immunity and on variations in preventative measures and policies such as social/physical distancing, reduced mobility (through staying at home, isolation, or quarantine), regular hand washing, wearing a mask, sneezing in the elbow rather than in open space or hands, avoiding meetings with large numbers of people, etc.) and (4) recovery process (either a certain time for recovery or a probability of recovery/death at each period, which may depend on the capacity to care for the infected, the timing of infection and the availability of possible treatments).

Based on these dimensions, the analysis of the epidemic simulation sample 1 and epidemic simulation sample 2 are presented at below. In the figures, the visual illustrations and the assembled code are shown in the simulation and the code construction panes, simultaneously.

Epidemic simulation sample 1

This model demonstrates the effect of staying at home on the rate and risk of the spread of the disease in an SIR model. The interactive simulation of the model shows the three classes of people: pink for the infected (sick and contagious), blue for the susceptible (healthy) and green for the recovered. The user may modify the parameter of the process of staying at home to visualize different scenarios. One scenario, of 10% of the population staying at home, is shown in Figure 4. The left figure is for an earlier time, but with the same parameters and initializations (see Figure 4)

Simulation 1's specifications. The simulation involves four parameters: (1) sick, (2) health, (3) time sick and (4) time to recover (see Figure 5).

The simulation starts with 1 sick individual and 100 healthy individuals. The initial locations of people are randomly determined (see Figure 6).

Initially the directions of the people who choose not to stay at home are randomly determined (between -30° and $+30^\circ$), they move three steps in these directions and their directions change randomly after every three steps (where the changes are between -30° and $+30^\circ$). Moreover, if they reach one of the edges while moving, their path bounces from the edge (see Figure 7).

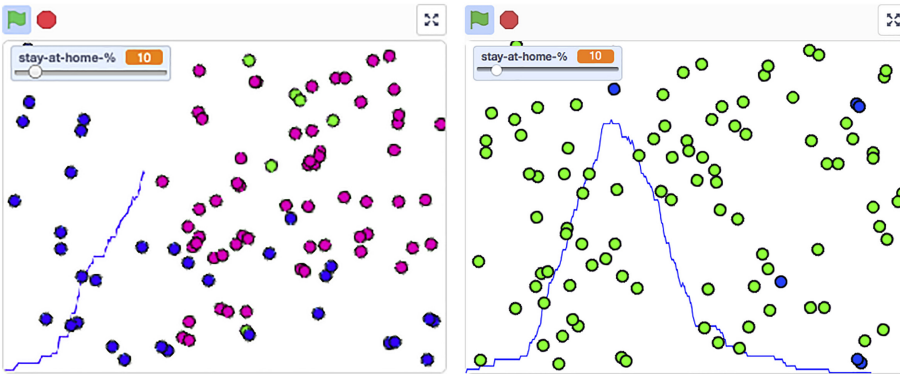


Figure 4.
Epidemic simulation
(Resnick, 2020)

Source(s): <https://scratch.mit.edu/projects/376656449/>



Figure 5.
The parameters in
simulation 1



Figure 6.
Initialization of the
parameters in
simulation 1

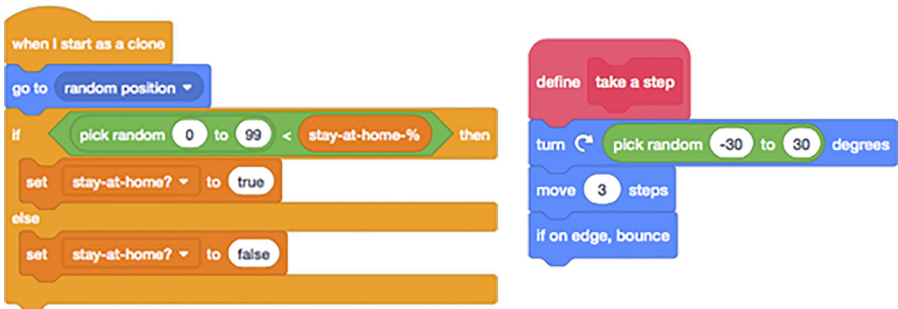


Figure 7.
Coding blocks for
illustrating movements
in simulation 1

When there is a contact between an infected person and a healthy person, the healthy person gets infected. This means the chance of contracting the disease per contagious contact is 1 (see Figure 8).

The infected people recover in a fixed amount of time (100 periods) (see Figure 9).

The number of cases over time, which is instantly updated according to the running simulation, is illustrated by a path traced by a character, referred to as the grapher in the code, whose motion is coded as an equation on an X-Y coordinate axes (see Figure 10).

Simulation 1's connection to mathematical concepts. Simulation 1 uses the following mathematical concepts: coordinate geometry for the location for the individuals and for the grapher; angles for the direction of the movement of individuals; mathematics operations of counting increments in the steps; the probability operation of picking a random angle for the



Figure 8.
Coding blocks for
illustrating
transmission in
simulation 1

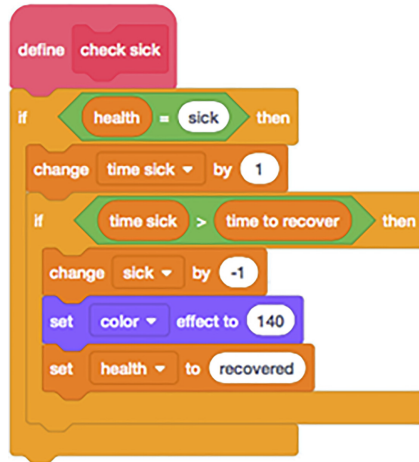


Figure 9.
Coding blocks for
illustrating the
recovery in
simulation 1



Figure 10.
Coding blocks for the
equation of the grapher
in simulation 1

direction of the turn and the number of people staying at home and an algebraic equation, which is a function of the sick individuals, for the path of the grapher.

Simulation 1's connection to CT concepts. Simulation 1 uses the following CT concepts: repetition and conditional logic, defining a block of blocks, parameters and changing looks of characters. By showing both a pictorial simulation and a graph of the variation in the number of people infected depending on people staying at home, it appears helpful with the basic understanding of the SIR models of growth in an outbreak and the effect of actions that may slow down the rate of spread. A student, especially if they have experience with coding simple games in Scratch, may choose to modify the code, for instance, to change the initialization such as time to recover to obtain rates closer to the real-life data of a pandemic.

Epidemic simulation sample model 2

Similar to simulation 1, simulation 2 demonstrates the effect of decreased movement (staying at home), hand washing and hospital capacity on the rate and risk of spread in an SIR model. The interactive simulation of the model shows four compartments of the population: Yellow for the susceptible, red or the infected, blue for the recovered (no longer contagious) and black for the removed (dead). It graphs the number of infected alongside the capacity of hospitals. To visualize the different scenarios, the user may change the parameters of four processes. One scenario for zero hand washing, maximum movement and a moderate hospital capacity is shown in Figure 11 at two different times.

Simulation 2's specifications. Simulation 2's specifications are as follows:

The simulation involves six parameters: (1) population, (2) initial rate, (3) recovery rate, (4) handwashing, (5) movement and (6) capacity. Users may adjust the hand washing and movement sliders to see how they affect the spread of infection or the capacity slider to change the healthcare capacity. Users may also press the spacebar to start/stop movement to mimic social distancing (see Figure 12).

The simulation starts with the cloning of individuals one by one; their initial locations are randomly determined, and each individual once cloned starts to move. A total of 50 individuals are created, where each of them has a positive probability of being sick (see Figure 13).

The directions of the people are randomly chosen, they move in these directions and the number of steps they move every period is determined by the chosen movement parameter.

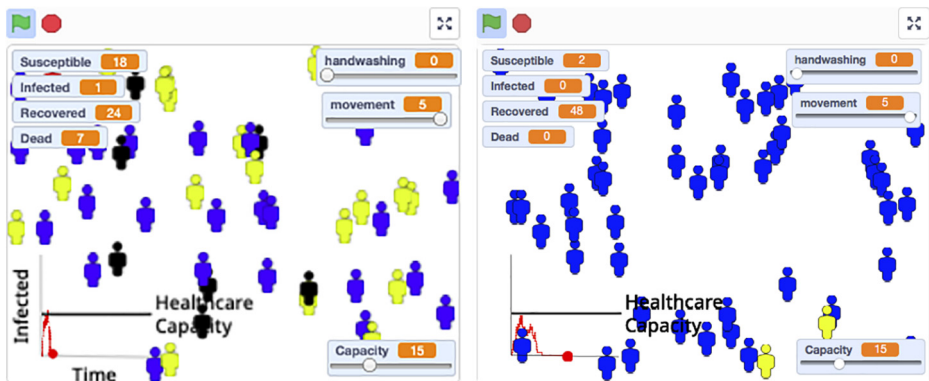


Figure 11. Infectious disease simulator with healthcare capacity (Brodie, 2020)

Source(s): <https://scratch.mit.edu/projects/380357960/>



Figure 12.
Coding blocks for
illustrating parameters
in simulation 2



Figure 13.
Coding blocks for
illustrating the
initialization step in
simulation 2

If people reach to one of the edges while moving, their path bounces from it, and this is the only source of change in moving directions (see Figure 14).

When there is contact between an infected person and a healthy person, the healthy person gets infected. The probability of infection is equal to $4(1.5 - \text{handwashing})/100$, which decreases proportionally to the chosen hand washing parameter (see Figure 15).

The infected people recover with a probability of 0.01 every period, and once the capacity of hospitals is exceeded, the infected people start to die with a probability of 0.01 every period (even if the number of infected people decreases below the capacity afterward). The simulation never stops, however, when there is no infected individual anymore, there will be no more change in the data (see Figure 16).



Figure 14.
Coding blocks for
illustrating the
movements step in
simulation 2

The number of cases over time is illustrated and this makes following the running simulation in real-time possible (see Figure 17).

The healthcare capacity is also illustrated, and it is assumed that infected people will not die as long as capacity is not satiated (see Figure 18).

Simulation 2's connection to mathematical concepts. Simulation 2 uses the following mathematical concepts: coordinate geometry for the location of the people pictographs and for the grapher; angles for the direction of the movement; mathematics operations of counting

Figure 15.
Coding blocks for illustrating the transmission equation in simulation 2



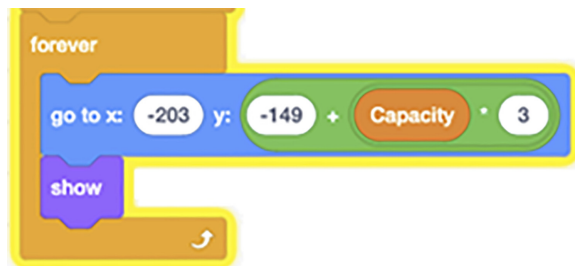
Figure 16.
Coding blocks for illustrating the recovery step in simulation 2



Figure 17.
Coding blocks for the equation of the marker on the grapher in simulation 2



Figure 18.
Coding blocks for the equation of the capacity sprite in simulation 2



increments in the steps of the infected, susceptible and time; the probability operation of picking each of the initial rates of recovery and death and an algebraic equation for the path of the grapher and of the health care capacity.

Simulation 2's connection to CT concepts. In simulation 2, the following CT concepts are used: repeat and conditional logic, defining a block of parameters and looks. This simulation shows the varied rates of infection hand washing, movement and capacity parameters. A student may choose to change the initialization steps, transmission equation or the movements of the people to closely mimic the data from a real pandemic.

The comparison of the two simulations

The basic dynamics of both simulations, simulation 1 and simulation 2, are based on SIR model, and when these discrete models converge toward continuous time models of large populations, they appear to behave like an SIR model (Kaplan, 2020). The first model is simple, as it only simulates the effect of staying at home, whereas the second model simulates the effect of three different factors, decreased movement, hand washing and capacity of hospitals. The second model also displays dynamic data of the parameter on the top left corner of the simulation pane. However, the movements are closer to reality in the first simulation, as the directions of people's movements are dynamically and randomly changing, unlike the movement of people always going in the same direction, except the bouncing at the edges, in the second simulation. Overall, both simulations illustrate the spread of disease and the effect of precaution on the process. In addition to changes that a user or reader of the simulation may make on the parameters that are modifiable, a user may change the initial states, steps or rates in the simulation to experiment with different scenarios.

Discussion

In this study, we pondered the ways in which certain presentations of SIR models have been helpful (or not helpful) in illustrating the dynamics of the outbreak, in demonstrating the responses needed to slow down the rate of the outbreak and in explaining the effects of certain responses to the public including students during the current global health crisis. In addition, we reflected on opportunities to use real-world problems experienced during the COVID-19 outbreak to promote elementary and middle grade students' computational and mathematical knowledge and skills.

Mathematical models which are used to explain real-world problems are not always transparent and some level of mathematical literacy required to comprehend them, however, CT tools would offer students better understanding of the problem and exploration of varied scenarios in the simulations. As demonstrated in our analysis, while differential equations used in the SIR model, such as $S' = -\beta IS/N$, require an understanding of high school academic and university mathematics, observing simulations and changing parameters in the Scratch simulations might require only a basic understanding of algebraic and computational concepts. Readers or users of the simulations, who may read and wish to remix the code, would require an understanding of assembling probability and algebraic expressions blocks in block-based programming as well as of computational thinking concepts of conditional statements, such as "if-then", "if-else-if", and the use of data block codes used to initialize parameters and to change them.

The two block-based simulations that we analyzed appear to provide some understanding of the SIR model of diseases, specifically the effect of precautions on the spread of the diseases. Given the mathematics required to manipulate these simulations, it appears that users might be motivated to re-examine the code to understand it and, if necessary, remix it. Advantages of the block-based programming languages are readability, ease of use, not

requiring an advanced level of technical coding knowledge and not encountering syntax errors (Abraham, 2019; Chumpia, 2018). The simplicity of Scratch simulations, however, makes it hard to model the reality of the pandemic and possible complex scenarios, and it is not possible to feed a Scratch simulation with real-life data, which is essential for accurate predictions. Significantly, these simple models could be useful as a basis for advanced simulations in text-based programming languages; once students are familiar with block-based programming, they quickly transition to understanding and building complex programs in text-based programming languages (Weintrop and Wilensky, 2015). After coding in Scratch, for example, students might pursue how to code similar and more advanced simulations in text-based languages, such as Python and Julia.

Text-based programming languages would ideally provide the complexity necessary for modeling and analyzing an outbreak based on scientific and statistical real-life data. Python, as a text-based programming language, allows its wide community of programmers and growing number of student coders to do more complex computations some of which may use real data to feed the model. Yeghikyan (2020), for instance, uses Wesolowski *et al.*'s (2017) SIR model to simulate the spread of COVID-19 in Yerevan, Armenia, using Python programming language. He is able to model the effect of mobility patterns on the diseases spread, by adding one more type of transition between the compartments of *S* and *I* that takes place due to the mobility of infected people from other locations to the location of interest. Similarly, Sargent and Stachurski (2020) use Atkeson's (2020) SIR model and publish a Python version of their code. Their simulations additionally show the parameters – including transmission rates, physical distances among people from different homes and lockdown of nonessential services – which may help slow down the spread, lower the peak or slow the peak of the outbreak. Julia is another programming language, which has been noted to be faster than Python in compiling big and complex codes, although its libraries are not as equally very well maintained as python. Vahdati (2019) provides a package in Julia for agent-based modeling (i.e. modeling of phenomena as dynamic systems of interacting autonomous agents) and its application on an SIR model for COVID-19. He simulates both the exponential growth that takes place if there is no intervention during the spread of COVID-19 and the growth with the effect of social distancing on “flattening the curve”, by making some agents simply not move, which he claims is a good approximation of reality.

Conclusion

Mathematics is an inevitable part of almost everything we do, and the most important thing about it is acquiring the ability to read the mathematics in our immediate environment. Following the COVID-19 outbreak, the society has been bombarded with the mathematical artefacts based on mathematical models of outbreak, which explain the rates and probabilities of spread, track progress of the outbreak and report the effects of the interventions on the pandemic. However, the complexity of mathematics makes it hard to comprehend the data and the dynamics behind these artefacts for nonexperts. Computational concepts when integrated with mathematics concepts in programming tools, such as Scratch, not only aid to illustrate the dynamics (e.g. parameters and the rates) of the outbreak but also demonstrate the effects of responses or controls needed to slow down the rate of the outbreak (e.g. the effects of certain precautions). The simulations include dynamic pictorial, numerical and graphical displays and offer an easy access to the code, which provides an opportunity to understand the basis of the recommended actions and policies during a pandemic while promoting mathematical and computational skills of learners.

As Resnick (2020) mentions, the coronavirus crisis presents many unprecedented challenges, but also some unexpected opportunities in the classroom settings. Implementing CT tools during this period of global health crisis not only offer better understanding for the

current health crisis but also have been helpful for understanding other interrelated crises in the post-pandemic world. Teachers may use the computational tools related to the COVID-19 pandemic to demonstrate to students the specific applications of their mathematical and computational knowledge and skills to tasks embedded in real-world contexts. The general context of disease-spread to teach mathematical modeling and computational thinking can be used as a springboard for empowering students to engage in more advanced simulations in text-based languages. Moreover, coupling mathematical models and computational thinking concepts could be helpful for students to realize the use of mathematics in reading, understanding and experimenting with simulations of magnitude, dynamics and recommended responses to facilitate informed decision-making during crises.

Notes

1. Source: <https://www.vox.com/2020/3/10/21171481/coronavirus-us-cases-quarantine-cancellation>.
2. Source: <https://scratch.mit.edu/>.
3. Source: <https://www.nctm.org/tools/pandemic2020/index.html>.

References

- Abraham, R. (2019), "Coding, programming, block-based, text-based what does it all mean?", available at: <https://www.teq.com/news/coding-programming-what-does-it-all-mean/> (accessed 20 May 2020).
- Adam, D. (2020), "Special report: the simulations driving the world's response to COVID-19", *Nature*, Vol. 580, pp. 316-318.
- Albrecht, J.R. and Karabenick, S.A. (2018), "Relevance for learning and motivation in education", *The Journal of Experimental Education*, Vol. 86 No. 1, pp. 1-10, doi: [10.1080/00220973.2017.1380593](https://doi.org/10.1080/00220973.2017.1380593).
- Atkeson, A.G. (2020), *What will be the Economic Impact of COVID-19 in the US? Rough Estimates of Disease Scenarios*, Working Paper [26867], National Bureau of Economic Research, March.
- Barclay, E. and Scott, D. (2020), "How canceled events and self-quarantines save lives, in one chart", available at: <https://www.vox.com/2020/3/10/21171481/coronavirus-us-cases-quarantine-cancellation> (accessed 22 April 2020).
- Brodie, I. (2020), "Infectious disease simulator with healthcare capacity", available at: [https://scratch.mit.edu/projects/380357960/](https://scratch.mit.edu/projects/380357960) (accessed 10 May 2020).
- Capitanelli, A. (2020), "Modeling the spread of diseases: a simulation exercise with SIR models", available at: <https://towardsdatascience.com/modeling-the-spread-of-diseases-821fc728990f> (accessed 12 May 2020).
- Chumpia, E. (2018), "Visual-based vs. Text based programming languages", available at: <https://www.codingkids.com.au/for-teachers/visual-based-vs-text-based-programming-languages/> (accessed 10 May 2020).
- Ciarochi, J. (2020), "How COVID-19 and other infectious diseases spread: mathematical modeling", available at: <https://triplebyte.com/blog/modeling-infectious-diseases> (accessed 20 April 2020).
- Daughton, A.R., Generous, N., Priedhorsky, R. and Deshpande, A. (2017), "An approach to and web-based tool for infectious disease outbreak intervention analysis", *Scientific Reports*, Vol. 7 No. 4607, pp. 1-10, doi: [10.1038/srep46076](https://doi.org/10.1038/srep46076).
- Froese, H. (2020), "Infectious disease modelling: beyond the basic SIR model", available at: <https://towardsdatascience.com/infectious-disease-modelling-beyond-the-basic-sir-model-216369c584c4> (accessed 20 April 2020).
- Gadanidis, G. (2017), "Five affordances of computational thinking to support elementary mathematics", *Education Journal of Computers in Mathematics and Science Teaching*, Vol. 36 No. 2, pp. 143-151.

- Giordano, G., Blanchini, F., Bruno, R., Colaneri, P., Di Phillo, A., Matteo, A. and Colaneri, M. (2020), "Modelling the COVID-19 epidemic and implementation of population-wide interventions in Italy", *Nature Medicine*, Vol. 26, pp. 855-860, doi: [10.1038/s41591-020-0883-7](https://doi.org/10.1038/s41591-020-0883-7).
- Haduong, P. (2019), "I like computers. I hate coding: a portrait of two teens' experiences", *Information and Learning Sciences*, Vol. 120 Nos 5/6, pp. 349-365, doi: [10.1108/ILS-05-2018-0037](https://doi.org/10.1108/ILS-05-2018-0037).
- Hoyles, C., Noss, R. and Adamson, R. (2002), "Rethinking the microworld idea", *Journal of Educational Computing Research*, Vol. 27 Nos 1-2, pp. 29-53.
- Kaplan, E.H. (2020), "COVID-19 scratch models to support local decisions", *Manufacturing and Services Operations Management*, Vol. 22 No. 4, pp. 645-655, doi: [10.1287/msom.2020.0891](https://doi.org/10.1287/msom.2020.0891).
- Kucharski, A. (2020), "Coronavirus: how maths is helping to answer crucial COVID-19 questions", available at: <https://www.newscientist.com/article/2233386-coronavirus-how-maths-is-helping-to-answer-crucial-covid-19-questions/#ixzz6MH6VRGr4> (accessed 12 April 2020).
- Lee, C.H. (2012), *Re-mastering the Master's Tools: Recognizing and Affirming the Life Experiences and Cultural Practices of Urban Youth in Critical Computational Literacy through a Video Game Project*, unpublished doctoral dissertation, University of California.
- Miller, S. (2020), "Computational thinking class enables students to engage in COVID-19 response", available at: <http://news.mit.edu/2020/computational-thinking-class-enables-students-engage-covid-19-response-0407> (accessed 10 May 2020).
- Namukasa, I.K., Gadanidis, G., Sarina, V., Scucuglia, R. and Aryee, K. (2016), "Selection of apps for teaching difficult math topics: concrete props, virtual applets and touch pad apps", in Moyer-Packenham, P. (Ed.), *International Perspectives on Teaching and Learning Mathematics with Virtual Manipulatives*, Springer, Dordrecht, pp. 275-300.
- Namukasa, I.K., Minakshi, P. and Miller, M. (2017), "Introductory CT and math in the middle grades: tools and resources", *Math + Code 'Zine*, Vol. 2 No. 2.
- National Council of Teachers of Mathematics (NCTM) (2020), "#KnowTheMath: impact of social distancing", available at: <https://www.nctm.org/blog/pandemicapp/> (accessed 20 April 2020).
- Obama, B. (2020), "If you're wondering whether it's an overreaction to cancel large gatherings and public events (and I love basketball), here's a useful primer as to why these measures can slow the spread of the virus and save lives. We have to look out for each other", [BarackObama] [Tweet], available at: <https://twitter.com/BarackObama/status/1238238576141352966> (accessed 12 May 2020).
- Ontario Ministry of Education (2020), *The Ontario Curriculum Grades 1–8 Mathematics Curriculum Context*, Queen's Printer, Ontario, ON.
- Resnick, M. (2020), "Scratch epidemic simulator", available at: <https://www.media.mit.edu/posts/scratch-epidemic-simulator/> (accessed 10 April 2020).
- Resnick, M. (2020), "Cultivating creativity during the coronavirus crisis", available at: <https://medium.com/mit-media-lab/cultivating-creativity-during-the-coronavirus-crisis-fedca3b6036> (accessed 12 April 2020).
- Resnick, M., Maloney, J., Monroy-Hernandez, A., Rusk, N., Eastmond, E., Brennan, K., Millner, A., Rosenbaum, E., Silver, J., Silverman, B. and Kafai, Y. (2009), "Scratch: programming for all", *Communications of the ACM*, Vol. 52 No. 11, pp. 60-67, doi: [10.1145/1592761.1592779](https://doi.org/10.1145/1592761.1592779).
- Rodrigues, H.S. (2016), "Application of SIR epidemiological model: new trends", *International Journal of Applied Mathematics and Informatics*, Vol. 10, pp. 92-97.
- Sanderson, G. (2020), "Exponential growth and epidemics", [3blue1brown], available at: <https://www.youtube.com/watch?v=Kas0tIxDvrg&feature=youtu.be> (accessed 2 May 2020).
- Sanford, J.F. (2013), "Core concepts of computational thinking", *International Journal of Teaching and Case Studies*, Vol. 4 No. 1, pp. 1-12.
- Sanford, J.F. and Naidu, J.T. (2016), "Computational thinking concepts for grade school", *Contemporary Issues in Education Research (CIER)*, Vol. 9 No. 1, pp. 23-32, doi: [10.19030/cier.v9i1.9547](https://doi.org/10.19030/cier.v9i1.9547).

- Sanford, J.F. and Naidu, J.T. (2017), "Mathematical modeling and computational thinking", *Contemporary Issues in Education Research*, Vol. 10 No. 2, pp. 159-168.
- Sargent, T.J. and Stachurski, J. (2020), "Modeling COVID 19", available at: https://python.quantecon.org/sir_model.html (accessed 12 May 2020).
- Shepherd, M. (2020), "Poor math literacy making it harder for people to understand COVID-19 coronavirus", available at: <https://www.forbes.com/sites/marshallshepherd/2020/03/23/is-the-math-too-hard-for-people-to-understand-covid-19-coronavirus/#344a229a6a9c> (accessed 2 May 2020).
- Siettos, C.I. and Russo, L. (2013), "Mathematical modeling of infectious disease dynamics", Vol. 4 No. 4, pp. 295-306, doi: [10.4161/viru.24041](https://doi.org/10.4161/viru.24041).
- Simler, K. (2020), "Outbreak", available at: <https://meltingasphalt.com/interactive/outbreak/> (accessed 20 April 2020).
- Skarky, B. (2020), "Math teacher uses COVID-19 to keep students learning during pandemic", available at: <https://www.robotlab.com/blog/math-teacher-uses-covid-19-to-keep-students-learning-during-pandemic> (accessed 20 April 2020).
- Skovsmose, O. (1994), "Towards a philosophy of critical mathematics education", *Educational Studies in Mathematics*, Vol. 27 No. 1, pp. 35-57.
- Stevens, H. (2020), "Why outbreaks like coronavirus spread exponentially, and how to 'flatten the curve'", available at: <https://www.washingtonpost.com/graphics/2020/world/corona-simulator/> (accessed 10 May 2020).
- Stroup, W.M. and Wilensky, U. (2014), "On the embedded complementarity of agent-based and aggregate reasoning in students' developing understanding of dynamic systems", *Technology, Knowledge and Learning*, Vol. 19 No. 1, pp. 19-52, doi: [10.1007/s10758-014-9218-4](https://doi.org/10.1007/s10758-014-9218-4).
- Tahiralli, J. and Ho, S. (2020), "COVID-19 curves: compare Canada and other key nations", available at: <https://www.ctvnews.ca/health/coronavirus/covid-19-curves-compare-canada-and-other-key-nations-1.4881500?cache=%2Fbuild-a-work-wardrobe-from-scratch-without-going-broke-1.2361542%3FclipId%3D89925> (accessed 10 May 2020).
- Taut, D. (2014), "Connecting maths to the real world", available at: <https://www.teachermagazine.com.au/articles/connecting-maths-to-the-real-world> (accessed 15 May 2020).
- Vahdati, A.R. (2019), "Agents.jl: agent-based modeling framework in Julia", *The Journal of Open Source Software*, Vol. 4 No. 42, p. 1611, doi: [10.21105/joss.01611](https://doi.org/10.21105/joss.01611).
- Walters, C.E., Meslé, M.M.I. and Hall, I.M. (2018), "Modelling the global spread of diseases: a review of current practice and capability", *Epidemics*, Vol. 25, pp. 1-8, doi: [10.1016/j.epidem.2018.05.007](https://doi.org/10.1016/j.epidem.2018.05.007).
- Wang, N., Fu, Y., Zhang, H. and Shi, H. (2020), "An evaluation of mathematical models for the outbreak of COVID-19", *Precision Clinical Medicine*, Vol. 3 No. 2, pp. 85-93, doi: [10.1093/pcmmedi/pbaa016](https://doi.org/10.1093/pcmmedi/pbaa016).
- Weintrop, D. and Wilensky, U. (2015), "To block or not to block, that is the question: students' perceptions of blocks-based programming", *Proceedings of the 14th International Conference on Interaction Design and Children*, ACM, New York, NY, pp. 199-208, doi: [10.1145/2771839.2771860](https://doi.org/10.1145/2771839.2771860).
- Wesolowski, A., zu Erbach-Schoenberg, E., Tatem, A.J., Lourenço, C., Viboud, C., Charu, V., Eagle, N., Engø-Monsen, K., Qureshi, T., Buckee, C.O. and Metcalf, C.J.E. (2017), "Multinational patterns of seasonal asymmetry in human movement influence infectious disease dynamics", *Nature Communications*, Vol. 8 No. 2069, doi: [10.1038/s41467-017-02064-4](https://doi.org/10.1038/s41467-017-02064-4).
- Wilkerson-Jerde, M.H. (2014), "Construction, categorization, and consensus: student generated computational artifacts as a context for disciplinary reflection", *Educational Technology Research and Development*, Vol. 62 No. 1, pp. 99-121.
- Wilkerson-Jerde, M. and Fenwick, M. (2016), "The practice of using mathematics and computational thinking", in Schwarz, C.V., Passmore, C. and Reiser, B.J. (Eds), *Helping Students Make Sense of*

the World Using Next Generation Science and Engineering Practices, National Science Teachers' Association Press, Arlington, VA, pp. 181-204.

- Wilkerson-Jerde, M., Gravel, B. and Macrander, C. (2015a), "Exploring shifts in middle school learners' modeling activity while generating drawings, animations, and computational simulations of molecular diffusion", *Journal of Science Education and Technology*, Vol. 24 Nos 2-3, pp. 396-415, doi: [10.1007/s10956-014-9497-5](https://doi.org/10.1007/s10956-014-9497-5).
- Wilkerson-Jerde, M., Wagh, A. and Wilensky, U. (2015b), "Balancing curricular and pedagogical needs in computational construction kits: lessons from the Delta Tick project", *Science Education*, Vol. 99 No. 3, pp. 465-499, doi: [10.1002/sce.21157](https://doi.org/10.1002/sce.21157).
- Wilkerson-Jerde, M.H., Shareff, R., Laina, V. and Gravel, B. (2018), "Epistemic gameplay and discovery in computational model-based inquiry activities", *Instructional Science*, Vol. 46 No. 1, pp. 35-60, doi: [10.1007/s11251-017-9430-4](https://doi.org/10.1007/s11251-017-9430-4).
- Wing, J. (2014), "Computational thinking benefits society", *40th Anniversary Blog of Social Issues in Computing*, Academic Press, New York, NY.
- Wing, J.M. (2017), "Computational thinking's influence on research and education for all", *Italian Journal of Educational Technology*, Vol. 25 No. 2, pp. 7-14, doi: [10.17471/2499-4324/922](https://doi.org/10.17471/2499-4324/922).
- Yates, C. (2020), "How to model a pandemic", available at: <https://theconversation.com/how-to-model-a-pandemic-134187> (accessed 15 April 2020).
- Yeghikyan, G. (2020), "Modelling the coronavirus epidemic in a city with python", available at: <https://towardsdatascience.com/modelling-the-coronavirus-epidemic-spreading-in-a-city-with-python-babd14d82fa2> (accessed 15 April 2020).

Corresponding author

Hatice Beyza Sezer can be contacted at: hsezer2@uwo.ca and

Immaculate Kizito Namukasa can be contacted at: inamukas@uwo.ca

For instructions on how to order reprints of this article, please visit our website:

www.emeraldgroupublishing.com/licensing/reprints.htm

Or contact us for further details: permissions@emeraldinsight.com