The essence of computational thinking and tools to promote it

Prof. Osman Yasar, State University of New York, Brockport

Osman Yasar is an endowed professor and director of the CMST Institute at The College at Brockport, SUNY. He established the first undergraduate degree program in computational science in the United States and developed a computational pedagogical content knowledge (CPACK) framework for teacher education. His research interests include engineering and science education, computational pedagogy, computational and scientific thinking as well as fluid dynamics, engine ignition modeling, and parallel computing. Yasar has a PhD in engineering physics and a MS in computer science and in nuclear engineering from the University of Wisconsin–Madison. He graduated with a BS in engineering physics and a MS in theoretical physics from Hacettepe University in Turkey. Contact him at oyasar@brockport.edu.

Dr. Jose Maliekal, The College at Brockport, State University of New York

Jose Maliekal is the Dean of the School of Arts and Sciences at The College at Brockport, State University of New York. He is an atmospheric scientist and have been teaching meteorology and climatology courses. His research interests include climate change and computational pedagogy.

Address: School of Arts and Sciences, The College at Brockport, Brockport, NY 14420 Phone: 585-395-5598 Email: jmalieka@brockport.edu

Dr. Peter Veronesi, The College at Brockport - SUNY

Peter Veronesi is an Associate Professor and program coordinator of science education for the Adolescence Science Education programs at The College at Brockport.

Dr. Leigh J Little, SUNY Brockport

Leigh Little received his doctorate in Mathematics from Arizona State University. He is currently a member of the Department of Earth Sciences at SUNY Brockport.
The essence of computational thinking and tools to promote it

Abstract

A decade of discourse to capture the essence of computational thinking (CT) has resulted in a set of skills whose teaching at the K-12 level poses many challenges because of the reliance on the use of electronic computers and programming concepts that are often found too abstract and difficult by young students. This article attempts to link cognition to basic computational processes, particularly modeling and simulation, that are known to facilitate deductive and inductive inquiries by scientists for decades. Empirical data from a quasi-experimental study in 15 secondary schools suggests a similar impact on student learning. This is consistent with learning theories that students learn science in the way that scientists think and work. In this article, we offer a viewpoint on the essence of CT and suggest that we teach students relevant cognitive habits prior to teaching them electronic CT skills. This will not only improve their critical thinking skills but also their motivation and readiness to learn electronic CT skills.

Keywords

Deductive and inductive thinking, cognitive processes, modeling and simulation

1. Introduction

A decade of discourse since the launch of computational thinking (CT) initiative by the computer science community has resulted in wide acceptance of the following relevant CT skill set for K-12 education:

- Abstraction and pattern generalization (including models and simulations)
- Systematic processing of information
- Symbol systems and representations
- Algorithmic notion of flow control
- Structured problem decomposition
- Iterative, recursive, and parallel thinking
- Conditional logic
- Efficiency and performance constraints
- Debugging and systematic error detection

While computational thinking has been recommended by the next generation science standards, obstacles remain in the way of integrating its practices into K-12 curricula. Since CT is linked to the use of electronic devices and equated with thinking by computer scientists, the discourse so far has not produced ways to separate it from computer programming. While the above skills may be appropriate for college freshmen, we argue that narrowing them down to a smaller and more fundamental set of cognitive competencies is necessary for the K-12 level. This requires a new approach in which we consider basic computation to be independent from the underlying hardware that performs it, be it electronic or biological. If we can identify skills and processes that are common to both electronic and biological computing (see Fig. 1), then through such a common skillset we might be able to link some of the above electronic CT practices to our typical cognitive (i.e., biological CT) skills. By promoting common core CT practices in K-12, we might not only improve fundamental cognitive skills of students but also prepare them for and motivate them towards learning additional electronic CT skills, including programming.
Common core CT processes: Seventy years ago, Alan Turing,\textsuperscript{24} widely recognized as the founder of computer science, suggested that if thoughts (i.e., information) can be broken up into simple constructs and algorithmic steps, then machines can add, subtract or rearrange them as our brains do. Electronic machines have since successfully taken many complex and voluminous computations off our brains, and further supported the view of brain as a computational device.

Basically, there appears to be two root causes of similarities between electronic and biological computing patterns. One of them, as Turing alluded to, is the invariant behavior of information. That is, information constructs behave either by uniting to make bigger constructs or breaking down to smaller ones. Devices that can track and tally this computable behavior (addition and subtraction) are called computers, regardless of their underlying structure. This duality of basic computation manifests itself in other higher level processes as we discuss it later.

Another cause may be the control and use of electronic devices by biological computing agents. Our use of an electronic device can certainly reflect the way we use our own biological computing device (i.e., our mind). Their utilization, however, depends on how we use them. So far, we have used electronic devices in various ways, including programming (text-based and visual), office work, communication, visual arts, video games, virtual reality, modeling and simulations. These range from easy tasks (e.g., automation of repetitive and voluminous work) to complex tasks (e.g., solving systems of differential equations for which there is no analytic solution). In almost all these cases, electronic devices basically follow preloaded instructions and therefore offer almost none or little cognitive insight to us, except that virtual reality and computer simulations are known to have generated some insight in scientific research.

In recent years, use of computer modeling and simulation has gone up because of their: a) accuracy to predict scientific phenomena, b) ability to conduct studies that are impossible to do experimentally due to size, access and cost, and c) economic impact to design, test, and manufacture industrial products such as engines, planes, cars, and new drugs.\textsuperscript{16} As a result, modeling and simulation is now regarded as a third pillar of doing science because it facilitates the deductive and inductive cycle of scientific thinking.\textsuperscript{6,12,16-17} Furthermore,
modeling and simulation has been found to support deductive and inductive approaches to teaching as well.20,22,27-35 So, judging from its utilization in both scientific research and teaching, one might say that modeling and simulation is a common process through which electronic computers and biological computing agents can resonate.

In this article, we attempt to build and examine a cognitive framework that can help us narrow down the CT skillset and promote it via readily available tools at K-12. In the sections below, we first offer a literature review of relevant concepts from neuroscience, cognitive sciences and epistemology. We will then explore how computational modeling and simulation tools (CMSTs) might help us link electronic and biological CT by designing a quasi-experimental study with empirical data from a diverse set of K-12 schools. Details of the study were given at earlier publications,27-32 but their relevance to cognitive skills is a new undertaking.33

2. Neuroscience and cognitive aspects of thinking

Many fields have their hands in the study of how learning takes place in the mind. Cognitive psychologists conduct empirical research into how people perceive, remember, and think. Developmental and educational psychologists form theories of human development and how they can be used in education. Neuroscientists use imaging techniques to understand the brain mechanisms that take part in learning. At the same time, cognitive and computer scientists form theories and models of the mind to study how computation generates cognition. In the sections below, we will review developments in each of these as they relate to our article.

Neuroscience of thinking: More than half a century ago, Donald Hebb,10 the father of neuropsychology, explained cognitive functions in terms of neural connections. Often referred to as “neurons that fire together wire together,” neural patterns became the center of our understanding of cognitive processing. According to Hebb, information is stored into the memory in the form of a specific pattern of neurons placed on a pathway and fired together. This view continues to dominate the field of neuroscience.2 Basically, the number and strength of neural pathways influence the storage and retrieval of information. Both the long-term storage and retrieval of information involve a synchronized distributed participation of neurons in related regions of the brain.13 As illustrated in Fig. 2, a memory or a newly learned concept can be a combination or outcome of previously formed memories and concepts, each of which might also involve another level of vast network of concepts and details mapped onto the brain’s neural network in a hierarchical way. When new information arrives, it lights up all related cues, neurons and pathways in a distributive process that is like the top-down action in Fig. 2, where new concept is broken up into related pieces. With the same token, retrieving a memory is a reassembly of its original pattern of neurons and pathways in an associative process like the bottom-up action in Fig. 2.

The key to storing a concept more permanently into the memory is, then, to link it to previously stored basic and retrievable concepts. And, the more links to associated concepts, the higher the chances of recalling this concept when needed later. Furthermore, cognitive retrieval practices attempted at different times, various settings and contexts is good because every time the recall is attempted it establishes more links that will help the remembering and learning. Exposure to new concepts, then, through links to multiple views from different fields of study is an effective retrieval strategy recommended by cognitive psychologists.2
Neuroscientists basically see little or no distinction between the acts of storage-retrieval, as described above, and the act of thinking. Similar to the distributive (top-down) process of storage, the mind attempts to interpret (i.e., think deductively) every new concept and information that it encounters in terms of previously registered models — objects, faces, scenarios, etc. And, as it grows further, the relationships among registered information eventually lead to interplay of various combinations and scenarios of existing models that eventually end up clustering, in an associative (bottom-up) fashion (i.e., thinking inductively), related details into conclusions, generalizations, and more inclusive models of information. The details our brain registers and stores and the hierarchical connections it establishes between them, along with these generalizations and conclusions, build over time like a pyramid-like structure in Fig. 2 that we have come to call mind. We often use software analogy to distinguish it from the brain.

**Computational theory of mind:** While the distributed structure of neurons and their connections (i.e., hardware) influence cognitive processing (i.e., software), the relationship between software (mind) and hardware (brain) is not a one-to-one relationship. According to computational theory of mind, our mind consists of a hierarchy of many patterns of information processing and, just as the case in electronic computing, these levels may range from basic computations to more complex functions (sequence or structure of instructions) and models (mental representations) of perceived reality and imaginary scenarios. While computational theory of mind has played an important role to separate mind from brain, the effort to model the mind as a rational decision-making computational device has not fully captured all our mental representations, particularly emotions. For example, the human mind is known for its energy-efficient operation, consuming as little electricity as a dim light bulb (20 Watts), while computational cognitive modeling and simulation of human brain is expected to consume $10^6$ times more electricity — equivalent to a nuclear power plant. One wonders, then, what accounts for the energy efficiency of human brain? Neuropsychologists, as well as evolutionary biologists, point to some structural (hardware) interference by an autopilot limbic system (animal-like brain) to by-pass, simplify, or reduce more elaborate cognitive functions of an evolved neocortex (outer parts of the human brain). It almost appears that we are caught up between two competing brains as illustrated by the top-down $\iff$ bottom-up cycle in Fig. 2: one that wants to simplify things and one that wants to dig things deeper. Cognitive scientist Read Montague points to some non-structural (software) tendencies to account for our brain’s energy-efficient operation. He suggests that concern for efficiency, as part of our survival, is a major driving factor. While this concern comes at the expense of being slow, noisy, and imprecise, it does assign value, cost, and goals to our thoughts, decisions and action. To assign these attributes, the mind carries out computations, builds models, and conducts hypothetical simulations of different scenarios. The human brain uses modeling not only for mental representation of external objects but also for its own internal computations so it can compare their values and costs before making decisions. Many argue that the uniqueness of human intelligence comes from the modeling of thoughts through language and, as we all know, thinking via language constructs appears to be a major difference between humans and other animals. As we see below from epistemology, modeling has been a core process of biological computing (i.e., thinking) from the dawn of humanity. So much so that we have invented electronic tools to enhance it.

3. **Modeling as an epistemological method**

Epistemology played an important role in establishing a philosophical basis for how we know what we know. While natural scientists laid a strong foundation in the 16th century for how to
acquire knowledge through empirical observations, philosophers continued their debate about the true source and nature of knowledge and validity of an observer’s subjective view of an objective world. This debate historically led to two opposing views, namely rationalism and empiricism. These opposing views are very similar to the dual aspects of human mind, which was mentioned earlier. Empiricism claimed that the mind is a blank slate and that it acquires knowledge a posteriori by putting together related pieces of information in a synthetic way using inductive reasoning. Knowledge acquired this way is not warranted because new experiences may later change its validity. Rationalism, on the other hand, has historically claimed that knowledge is acquired a priori through innate concepts. Innate knowledge is warranted as truth, and decisions and conclusions can be derived from it in an analytic way using deductive reasoning.

Immanuel Kant11 argued against the views of both rationalists and empiricists and combined the two to lay the foundations of epistemology. He recognized what experience brings to mind as much as what mind itself brings to experience through structural representations. In applying mathematical, logical, and physical representations to study of nature, Kant considered that knowledge developed a posteriori through synthesis could become knowledge a priori later. Furthermore, according to him, a priori cognition of the scientist continues to evolve over the course of science's progress.19 The epistemological method Kant established more than two centuries ago has been the method by which scientific knowledge has evolved, though its dual (deductive/inductive) cycle of change has been slow until recent years.12 Historically, once proven or validated, a hypothesis or an observation was revisited at a slow pace, sometimes spanning generations, because of both limited resources (time, money, equipment, etc.) and the overwhelming number of other questions begging for an answer or proof. However, the growing scientific knowledge and the number of researchers tackling a problem as well as the increasing capacity of technology have all now shortened the timescale of scientific progress. Concepts and theories which were once considered true and valid are now quickly being modified or eliminated.

Modeling has been an important tool for scientific research for hundreds of years. It works exactly as articulated by Kant, by which a previously founded theory or concept is analyzed first and then synthesized – after testing, sorting, and updating – to either validate or change it. This process of often called conceptual change.25 As illustrated in Fig. 2, scientists often start with a model of reality based on current research, facts, and information. They test the model’s predictions against experiment. If results do not match, they, then break down the model deductively into its parts (sub models) to identify what needs to be tweaked. They retest the revised model through what-if scenarios by changing relevant parameters and characteristics of the sub models. By putting together inductively new findings and relationships among sub models, the initial model gets revised. This cycle of modeling, testing, what-if scenarios, synthesis, decision-making, and re-modeling is repeated while resources permit until there is confidence in the revised model’s validity. In the past several decades, computers have accelerated this cycle because not only they speed up the model building and testing but also help conduct studies that are impossible experimentally due to size, access and cost.

Conceptual change has been an important element of scientific progress – both at the community level in the form of a paradigm shift12 and at the individual level as observed in laboratory environments.6 It is an element of ordinary thinking and learning by all,25 yet not everyone uses its dual (deductive/inductive) reasoning cycle as consistently, frequently, and methodologically as scientists. Since learning sciences suggest that students should learn
4. The essence of computational thinking

The presence of duality in matter and mind has been noted for ages. One might argue that heterogeneity is the essence of duality because heterogeneous things – both matter and information – appear to behave in one of only two ways: they either unite to form bigger ones or break down to smaller ones. With the same token, the essence of heterogeneity itself is this dual (associative and distributive) behavior of discrete stuff. So, the degree of heterogeneity, its growth and degradation (breakdown) all depends on its overall dynamics of computable actions. In a way, formation of heterogeneity is akin to the act of modeling – an associative (inductive) action by which related parts are added together to make a whole. Such action is either driven by external forces or by a collective “trial and error” process driven by various conditions and rules of engagement — much like a simulation.

Our brain’s natural inclination to process information in an associative/distributive fashion, and to store and retrieve information in a scatter/gather way seems to be a manifestation of the basic duality engrained in the fabric of matter and information. This inclination may just be an evolutionary response, shaped up over many years, to optimize the handling of incoming sensory information whose quantifiable nature resonates with distributive and associative operations. Accordingly, one can argue that associative processing of information by a computational mind is the essence of inductive reasoning (i.e., abstraction), through which details are put together, focus is placed on general patterns, and priority and importance are assigned to newly acquired information. Learning scientists and cognitive researchers point to abstraction as an inductive process that helps us simplify, categorize, and register key information from sparse, noisy, and ambiguous data for quicker retrieval and processing. With the same token, distributive processing of information appears to be the essence of deductive reasoning, through which a general concept is analyzed and broken down in terms of its possible constituencies and their applicability and validity. It helps us decompose a complex issue by dividing (scattering) the complexity into smaller pieces and then attacking each one separately until a cumulative solution is found (gathered).

Associative (+) and distributive (-) processing, storing, and retrieval of information appears to be the essence of thinking generated by a computational mind. We put this dual property at the core of information processing by both electronic and biological computing devices (Fig. 3). What follows next is higher-level deductive/inductive cognitive processes, whose iterative and cyclical usage will depend on the underlying device structure and the quality and quantity of the environmental input it receives. Some elements of the electronic CT skill naturally overlap with these high-level cognitive (biological CT) processes. Others can be considered as device-dependent skills resulting from the use of an electronic device for problem solving by a biological computing agent.

Of all the characteristics of the electronic CT skills listed in the literature, the two fundamental ones that have a correspondence in biological CT, as we defined here, are abstraction and decomposition. If abstraction is considered the packing (modeling) of things, then decomposition is the unpacking and examination of its contents. We all live in a
constant cycle of packing and unpacking of information based on our changing need for details and generalizations. This process is akin to the top-down ⇔ bottom-up cycle in Fig. 2. However, while everyone uses abstraction and decomposition, not all are equally aware of the importance of these two essential biological CT skills, nor are we all practicing and utilizing them fully and equally.

We argue that figures 2 and 3 collectively capture the essence of both computational and scientific thinking – in fact, they are equal in a scientist’s mind. Teaching young minds an awareness about the role of computation in cognition as well as a cognitive habit of conceptual change might help them think like a scientist and be prepared to use electronic computing devices to further such thinking regardless of whether they work as a scientist or not. The following section examines our cognitive framework in terms of student learning. We hypothesize that statistically significant evidence should validate our cross-disciplinary view.

5. Technological and pedagogical tools to support CT education

We ran a 5-year (2003-2008) multi-faceted experiment to study pedagogical aspects of modeling and simulation. The research involved a quasi-experiment study of treatment and control groups. A multi-tier teacher-training program was offered to support the use and testing of modeling tools listed in Table 1 in both classroom and after school settings. These menu-driven tools allow students to quickly set up and run a model using an intuitive user interface with no knowledge of equations, scientific laws, and programming or system commands.

Table 1: A typical list of user-friendly modeling and computer simulation tools

<table>
<thead>
<tr>
<th>Tool</th>
<th>Description</th>
<th>Website</th>
</tr>
</thead>
<tbody>
<tr>
<td>Excel</td>
<td>constructs hands-on modeling &amp; simulations using rate of change (new = old + change).</td>
<td></td>
</tr>
</tbody>
</table>

Assuming a positive relationship between teacher variables (knowledge and ability) and student outcomes (knowledge, ability, and interest), we explored the impact of three independent variables (technology, pedagogy, and training) on teaching and learning. The training included 80 hours of technological knowledge (TK), 80 hours of technological content knowledge (TCK, teaching content through technology), and 40 hours of technological pedagogical content knowledge (TPACK, teaching content through technological and pedagogical tools). More than 300 in-service math and science teachers from 15 secondary schools (grades 7-12) took part in initial TK training of this initiative. About half of teacher participants moved on to the 2nd year and about half of those moved on to the 3rd year TPACK training. Interdisciplinary aspects of TPACK have been addressed in...
many studies that can be found at http://www.tpack.org. Our focus is on computational technology, a subset of TPACK.

A mixed-methods approach was used to collect and analyze qualitative and quantitative data, including interviews, activity logs, observations, Likert-scaled surveys, artifacts, report cards, test scores, and standardized exams. Results of this comprehensive study have been published already and here we will only briefly review relevant findings. Annual surveys of teachers showed that utilization of CMSTs in the classroom was directly linked to the amount of teacher training. For example, while only 60% of the teachers reported occasional use of CMSTs in their classrooms after one year of training, 78% reported regular use after three-year training. Besides teacher confidence and comfort level with the tools, student engagement, grade levels and subject areas also affected the intensity of CMST utilization in the classroom. A typical annual survey, shown in Table 2, indicated that the higher the grade level the more regularly the tool usage. Modeling is a common practice in math but it may not need as many resources as science classes to simulate time-dependent dynamics of scientific phenomena. By the end of the initiative, we developed a large database of modeling-based curricular modules and lesson plans to increase utilization by participating math and science teachers. Currently they are well utilized, reaching 80-100 downloads per day by educators around the world (see http://digitalcommons.brockport.edu/cmst_institute/).

Annually, about 5,000 students had a chance to benefit from modeling-based deductive and inductive teaching and learning during the initiative. Teachers reported assessments based on their own baseline data of the same classes as well as the unit tests given before and after the use of modeling-based teaching. Professional evaluators triangulated these findings through their own classroom observations. In a typical survey conducted by external evaluators, most teachers agreed that using modeling tools in their classrooms significantly increased student engagement. 97% of mathematics and 92% of science teachers using modeling in the classroom agreed that use of such tools made subject-related concepts significantly more comprehensible. Furthermore, 83% of science teachers and 76% of math teachers reported that it led to even a deeper understanding of STEM concepts. 100% of technology, 72% of math, and 31% of science teachers reported observed improvement in students’ problem solving skills. Student reaction to modeling (versus traditional techniques) was found to be 97% favorable in math and 77% in science classes. While science classes utilized technology less due to limited access and lack of science-related modeling examples, in instances where it was utilized, a deeper understanding of science topics was achieved, compared to math topics (83% vs. 76%). Students in higher-grade levels found computational modeling more engaging in both math classes (grades 7-8: 77% vs. grades 9-12: 90%) and science classes (grades 7-8: 75% vs. grades 9-12: 85%). Modeling was even found helpful to non-traditional (special education) learners: again, the higher the grade levels the higher the engagement – i.e., math classes (grades 7-8: 76%, grades 9-12: 100%) and science classes (grades 7-8: 75%, grades 9-12: 85%).

An upward trend in student achievement data (report cards and standardized tests) was also noted in participating urban and suburban districts during the 5-year study. Students were coded and followed by evaluators until their graduation to monitor progress as a function of
multi-year instruction from CMST-enhanced teaching. The percentage of students receiving a State Regents diploma increased significantly from the baseline (RCSD: 21% → 59%, BCSD: 84% → 95%). The passing rate (>65/100) in State Grade-8 Math exam increased in RCSD from 10% to 33%, while the passing rate in State Regents Math-A exam (Grade 11-12) increased from 13% to 67%. Passing rate in sciences increased in areas in Physics (3% → 22%) and Chemistry (9% → 27%). At BCSD, passing rates improved in mathematics (Math-A: 51% → 99%) and sciences (Physics: 52% → 78%). The number of students taking General Physics at BCSD increased from 50% to ~100% and the number of students taking AP Physics also doubled. Student passing rates at both districts seemed to reflect relative participation of district’s math and science teachers in the study. All the improvements were found to be statistically significant (p ≤ 0.01) for the sample sizes from each district.

An inductive analysis of open-ended questionnaires by two independent coders helped several major themes to emerge. One of these is that different genders responded differently to teaching through technology. While male students showed more interest in playing with technology tools and plowing through the details deductively (top-down arrows in Fig. 2) with less regard to the big picture, female students initially seemed reluctant and timid but excelled when details were conceptualized inductively (bottom-up arrows in Fig. 2) and linked to real-world problems, projects, and applications. The difference in gender response was triangulated with quantitative results. While cohorts of the 8th grade male and female students from both districts had a gap in their average math performance at the beginning of the initiative, not only were the gaps closed but also reversed four years later as shown in Table 3. At RCSD, while both male and female students did much better than four years earlier, the graduation rate of the same cohorts still reflected a gender-based trend in performance growth, favoring female students. The sample sizes for male and female students were roughly the same at both districts, with about 1200 at RCSD and 150 at BCSD. Two-proportion z-scores indicate that the difference is statistically significant. The column p indicates the confidence level that the difference between males and females may be due to a nonrandom effect. This is consistent with gender-based research reported by Repenning involving use of AgentSheets in K-12.

Table 3: Gender-based response at urban (RCSD) and suburban (BCSD) schools.

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Male</td>
<td>Female</td>
</tr>
<tr>
<td>RCSD Math Cohort</td>
<td>13%</td>
<td>10%</td>
</tr>
<tr>
<td>Graduation Rate</td>
<td>34%</td>
<td>44%</td>
</tr>
<tr>
<td>BCSD Math Cohort</td>
<td>92%</td>
<td>84%</td>
</tr>
<tr>
<td>Graduation Rate</td>
<td>85%</td>
<td>90%</td>
</tr>
</tbody>
</table>

Several treatment-control comparisons were conducted. One of these included a pair of treatment and control teachers from the same school teaching properties of quadrilaterals in a mathematics class. The target teacher used GSP in a class of 24 pupils while the control teacher used conventional methods in a class of 14 pupils. Both teachers conducted the same unit test. Even though the CMST teacher taught a more crowded class, his classroom average was 82.5 versus 49.5 for the other class. The second study involved a math triathlon involving use of advance graphing calculators to solve questions from State Regents Math A and B tests. Scored by external judges, including teachers and college faculty, this study revealed that students taught by CMST teachers outperformed other students in all categories: Math-A: 60.26 vs. 49.54; Math-B: 71.9 vs. 55.6; and 7-8 Grade Math: 64.0 vs. 58.6.
An after-school project challenge provided students more time and freedom than a regular classroom setting to apply, test, and revise the constructed computational models. Annually, more than 200 students had a full semester to develop 4-person team projects. Scoring rubric included problem statement, application of the model to a problem of interest, data analysis, teamwork, originality, electronic demonstration, and presentation of the results before a panel. Extra points were given for use of multiple tools, demonstrated understanding of computational, mathematical and scientific content, level of challenge, and knowledge and skills demonstrated beyond team’s grade level. The incentives helped push students to go beyond initial job of model construction, playful experimentation, and introductory exposure to STEM concepts. As we expected, when used together modeling and simulation appears to afford students a constructivist opportunity to cycle iteratively back and forth between the inductive and deductive approaches to learning.27,31

Experience by a group of 9th high school students from this project-based program, as reported in their own articles,34-35 offers a testimony of how students can gain a deeper understanding of scientific content using modeling tools. Figure 4 shows harmonic motion of a box attached to a spring, using one of the tools, the Interactive Physics (IP). The following year, these 10th graders inquired about operational principles of the tools they used. Using a simple rate of change equation, they replicated the results they had found earlier with IP. Figure 5 shows the same harmonic motion using the “new = old + change” rate of change equation to compute for position and velocity profiles via Excel. The progression by these students show that the learner can start either with a readymade model, or construct one using a pull-down menu, that represents the scientific phenomenon under study and conduct fun experiments without having to know the details of the model and the laws that govern its motion. If it stops there, then we can say that the top-down deductive approach has engaged students in STEM activities. But, if the learner is tempted to continue and inquire about the initial model’s constitutive parts and forces that act on them, then he can run simulations by changing characteristics of the parts and forces to inductively construct a new model and physical setting that better represent the reality. This cycle can be repeated until the desired knowledge or outcome is reached. This deductive and inductive way of learning, through inquiry and experience, is nothing but how scientists do their work.1,5,15

In this process, students also get a chance to realize the virtue of core CT skills such as decomposing a
problem into smaller chunks and correlate the cost of computation and the choice of tool to the needed accuracy. Figure 5 shows velocity and position profiles for various degrees of decomposition: *the finer the granularity (smaller dt), the more accurate the answer.* While a human can calculate a few data points by hand when \( dt \) is manageably small, the need for automation (and accuracy) becomes obvious for much smaller \( dt \) values. So, Excel can be used to automate the calculation and graph the curves as shown in Fig. 5, but for much smaller step sizes, students discover that it cannot help process millions of data points in computationally intensive cases. The need for finer, faster, and more accurate automation, via programming (such as Python) becomes self-evident. In short, contrary to concerns that focusing on modeling might keep learners from more mainstream computing practices, our experience shows that it instead motivates them to learn and use computer programming.

6. Conclusion

We presented a view on the essence of computational thinking by separating it from electronic computing devices and programming practices. We base this view on the following premises. We consider heterogeneity to be the essence of a dual (*associative* and *distributive*) behavior that is often computable (via *addition* and *subtraction*) by a tallying device (biological or electronic). Accordingly, an iterative and cyclical dynamics of a set of dual (associative/distributive) processes appears to be a common mechanism by which all quantifiable stuff forms and grows, including matter and information. We have come to call this dual process “modeling and simulation” in the context of information processing, but it should be commonly and equally applicable to natural dynamism of all other discrete forms.

Many years of experience with using modeling and simulation tools in research has shown that such tools are significantly aiding scientific progress because they facilitate the deductive and inductive cycle of the scientific method. The latest software tools have made such experience possible for students as well by allowing them to set up and run a model with no knowledge of equations, scientific laws, and programming. Empirical data supports this as briefly presented in this article. Presented findings are consistent with a growing body of research that identifies computer simulation as an exemplar of inquiry-guided (inductive) learning through students’ active and increasingly independent investigation of questions, problems and issues.

Modeling and simulation carries a constructivist pedagogy whose iterative and cyclical nature mirrors Kant’s epistemological method represented in Fig. 2. Basically, modeling provides a general simplistic framework from which instructors can deductively introduce a topic without details, and then move deeper gradually with more content after students gain a level of interest to help them endure the hardships of effortful and constructive learning. This deductive approach takes away the threatening and boring aspect of STEM learning. Simulation, on the other hand, provides a dynamic medium to test the model’s predictions, break it into its constitutive parts to run various *what-if* scenarios, make changes to them if necessary, and put pieces of the puzzle together *inductively* to come up with a revised model. It provides a dynamic medium for the learner to conduct scientific experiments in a friendly, playful, predictive, eventful, and interactive way to test hypothetical scenarios. This inductive process enables the learner to put pieces of the puzzle to come up with a revised model. Anyone who learns in this fashion would, in fact, be practicing the craft of scientists.

Finally, we conclude that if modeling and simulation is an essential element of biological CT, then it ought to be an essential element of electronic CT skillset as well. There needs to be an
effort to add this view to the discourse by the community. While the information revolution has taken electronic computing devices to every corner of the globe, few are familiar with and relate to computational modeling and simulation. In fact, even some researchers and educators consider computational modeling as \textit{ad hoc} technology. Furthermore, the field of computer science is not generally considered as a branch of science because it supposedly deals with artificial phenomena, not natural phenomena. We disagree. The universality of modeling and simulation, as a representative of (computational) behavior of matter, should indeed change this and put computing at the heart of sciences. While scientific revolutions of Einstein and Darwin seem to have been much realized, that computational revolution started by Turing may be how finally our knowledge can come together to make more sense.

\textbf{Acknowledgements}

Support from the National Science Foundation, through grants EHR 0226962, DRL 0410509, DRL 0540824, DRL 0733864, DRL 1614847, and DUE 1136332, is greatly appreciated.

\textbf{References}