Integrating Interactive Computer Simulations into K-12 Earth and Environmental Science

Michelle Zhu, Nicole Panorkou, Pankaj Lal, Sowmith Etikyala, Erell Germia, Pricilla Iranah, Bharath Samanthula and Debasmita Basu
Montclair State University, zhumi, panorkoun, lalp, etikyalas1, germiae1, iranahp1, samanthulab, basud2@montclair.edu

Abstract - **This paper discusses our work in progress aiming to explore how computer simulations can be integrated into the K-12 curriculum of Earth and Environmental science. Several interactive simulations using Netlogo, a multi-agent modeling environment, and Scratch, a visual programming software are being developed with steerable parameters and the corresponding output plots for students to manipulate and interpret the results, respectively. Here, we present two simulations we designed on water cycle and discuss how these may help students learn about the distribution of water and its continuous move in the ecosystem.**

Index Terms – Computer Simulations, Earth and Environmental Science, Netlogo, Scratch

INTRODUCTION

Earth and Environmental Science (EES) studies the earth's dynamic history and aims to address the emerged environmental challenges because of human activities. It offers a rich context for investigating STEM education as it is a multidisciplinary field integrating almost every discipline of science, such as physics, biology, chemistry, geology, ecology, mineralogy, and oceanology. Our goal in this study is to develop students' understanding of EES and introduce the next generation of citizens to the importance of utilizing the environmental resources in a sustainable way and studying its profound impacts on global society [1].

To do that, we introduce and instill computational thinking [2] into EES studies since computational science, such as mathematical modeling, dynamic parameterized simulation, parallel computing, data generation and analysis as well as scientific visualization, can assist in modeling these complex earth and environmental phenomena and concepts and facilitate improved learning and retention. We consider computational thinking to be the tool that can help students translate their novel ideas pertaining to EES into action by engaging in logical reasoning, critical thinking and problem solving activities. Therefore, we developed several simulations for EES topics such as the water cycle and the solar system using techniques such as parameter steering, attractive animation and rich input and output plots. Additionally, due to the current technology trend, we also plan to expose our students to "deep learning" technology. Deep learning is a subarea in machine learning that utilizes neural networks to capture the characteristics of training datasets and predict the desired features for testing datasets.

We are currently designing modules for grades 5-7 based on New Jersey curriculum and building on the features elaborated above. Each module is designed around an EES theme (e.g. water cycle) and is based on a seamless integration of computer science, mathematics, and environmental science disciplines aiming to expose students to interdisciplinary approach of gathering knowledge and skill development. We plan to cover the following Next Generation Science Standards (NGSS): 5-PS2-1, 5-ESS1-2, 2-2, 3-1, MS-ESS1-1, 1-2, 2-1, 2-3, 2-4, 2-5, 2-6, 3-3,3-4.

We include a strong mathematical component in our modules to provide opportunities for students to see the *purpose* and *utility* of mathematics [3] in EES and computer science. The curriculum modules' design aims to incrementally develop students' EES, computational and mathematical thinking skills and follows the Next Generation Science Standards, the Common Core State Standards for Mathematics and the New Jersey student standards.

For our design, we considered "*learning by doing*," a pedagogy that is often attributed to the work of Dewey [4] and that can also be seen as rooted in Constructionism [5,6], a variant of Constructivism [7] where learning is seen to be especially felicitous when its goal is to produce a shareable artifact. *Learning by doing* projects are compatible with a *project-based learning* and an *inquiry-based learning* instructional approach. By having the EES as the umbrella, our goal is to emphasize that it is not just about "using" or applying math and computer science to EES; it is about "utilizing" computational and mathematical thinking to facilitate critical thinking skills around EES concepts, and design, create and share artifacts that would express and challenge views on this topic.

In this paper, we present an example of our curriculum design which focuses on the water cycle. We describe two simulations we designed using Netlogo and Scratch and discuss how these can be used with grades 5-7 students. The design is currently undergoing incremental improvements and the final products will be released for community use.

WATER CYCLE SIMULATIONS

From the early years of education, students learn of the importance of water as a resource that helps to sustain life on Earth. In current curricula, students learn about the water cycle's processes of evaporation, condensation and precipitation through pictures and illustrative videos. In contrast, our team aims to explore this through a question: How may we design more interactive instructional tools that would allow students to actively engage with and meaningfully construct this content?

I. Water Cycle in Netlogo

Netlogo is a multi-agent programmable modeling environment that can be used to design dynamic simulations of various complex phenomena [8]. The animated outputs and result plots of Netlogo can help students understand the evolving dynamics of the interactive simulations and the phenomena they model. The Netlogo simulation we designed illustrates how the temperatures at ocean and mountain, humidity and various terrains, including forest, desert and city, can affect the water cycle. There is a fixed number of water molecules in this simulation, so that we can keep track of each molecule and make proper distribution of the water on the different parts of the earth including ocean, ice, glacier, surface water and underground water. The purpose of this simulation is for the students to be able to modify these parameters in order to purposely create phenomena, such as more rain or less snow in the mountain. Students are also able to trace the path of a single molecule in the cycle using a toggle button. To achieve a better visual effect, the speed of evaporation is correlated with the temperature and humidity. In addition, the lake water level reflects the water amount left after evaporation.

In the simulation, all water molecules cycle through different parts of earth within the ambit of virtual ecosystem that we created. First, the water molecules' journey begins in the lake/ocean, where the water gets evaporated and forms a cloud. The flowchart of the code is presented in Fig. 1, which shows the general structure of our program, and the code segment in Fig. 2 demonstrates how the evaporation and the clouds are formed from simulation parameters. The color of the water molecules allows us to visualize the transition appropriately. The color first starts with red and then converts into white as the water molecules form a cloud. The cloud is then transported to the mountains and the precipitation process begins when the cloud turns grey. When the humidity is high, the speed of evaporation slows down.

Figures 3 and 4 illustrate the screenshots of the most recent version of our Netlogo simulation. Figure 3 shows an ocean and mountain terrain while figure 4 depicts a desert terrain. Students may use this simulation for data collection, interpretation and analysis by generating an output plot with input and output variable values. For instance, students may figure out the relationship between various input and outputs based on their exploratory experiments within the simulation. The Penman formula for the evaporation rate ($mm\ day^{-1}$) from a lake can be simplified to the equation:

$$E_{mass} = \frac{.000_1}{100-A} a\ 15(T - T_8)(80 - T)$$

where $T_m = T\ a\ 0.006\ h$, h is the elevation in meters, T is the mean temperature, A is the latitude in degrees and T_8 is the mean dew-point [9]. The evaporation rate approximately has a linear relationship with the temperature.

Through the exploration of the tools of the simulation, students develop hypotheses, formalize ideas, and analyze the results in a critical way [10]. In terms of mathematics, they understand the idea of a variable as a quantity that varies, and they reason about the relationships between the co-varying quantities by comparing visual, graphical and algebraic representations.

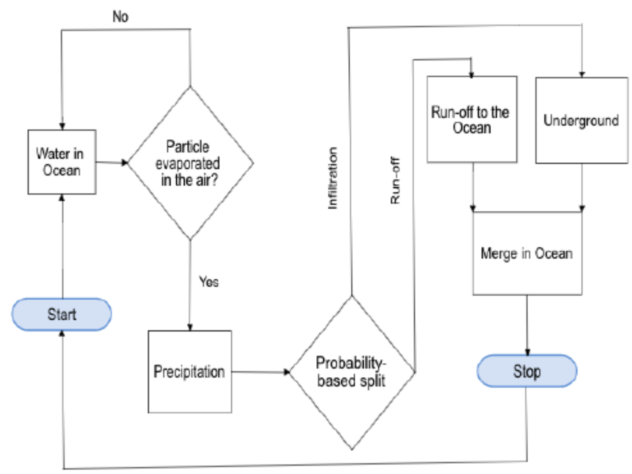

FIGURE 1
THE FLOWCHART OF OUR NETLOGO SIMULATION

```
to try-vapor
  ask vapors[
    if (color = red + 2)[
      if (random(humidity) mod (humidity / 15) = 0 and ycor = surface)
      [
        set ycor ycor + (temperature / 10)
      ]
      ifelse (ycor + (temperature / 10) < 35) [
        if (ycor != surface)
        [
          set ycor ycor + (temperature / 10)
        ]
    ]
    [
      set hitCloud? true
      set color white - 1
    ]
    ]
  ]
    if (hitCloud? = true)[
    ask clouds[
            set size size + (70 / 30)
            set hitCloud? false
      ]
    ifelse (any? vapors with [color = red + 2 or color = blue + 0.5 or
      color = blue - 0.5 or color = blue + 1])
        [
        ]
        [
          ifelse (climate = "temperate")
          [
        ask clouds[
            set color white
        ]
          ]
          [
        ask clouds[
            set color grey
        ]
          ]
        ]
    ]
end
```

FIGURE 2
THE CODE SEGMENT FOR WATER CYCLE SIMULATION IN NETLOGO

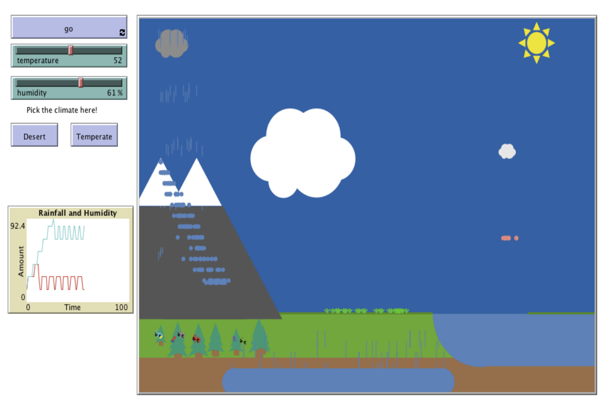

FIGURE 3
THE NETLOGO INTERFACE OF THE WATER CYCLE WITH OCEAN AND MOUNTAIN TERRAIN

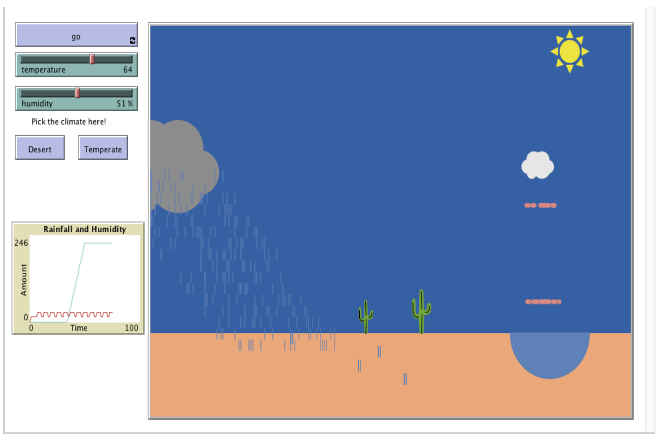

FIGURE 4
THE NETLOGO INTERFACE OF THE WATER CYCLE WITH DESERT TERRAIN

II. Water Cycle in Scratch

Scratch is a visual, drag-and-drop block programming language that allows young users to learn computer programming while working on the creation of interactive, media-rich projects [11]. The goal of Scratch is for students to learn how to express their ideas through computer programs. The simulation we designed in Scratch illustrates how a single water molecule moves around the water cycle. We created two loops for this cycle: In the first loop, the water molecule evaporates from the lake into the air, forming a cloud. Then it is condensed into liquid water and falls back to the lake in the form of rain (Fig. 5). In the second loop, after the water molecule evaporates, it travels to the mountains and during precipitation it falls in the form of snow (Fig. 6). Figure 7 presents blocks used to design the movement of the water molecule. Similar codes are designed for the movement of the other components of the water cycle, such as the clouds and rain.

We designed two types of investigations with this simulation. In the first investigation, students watch the simulation and then are asked to get to know the code by exploring what each block can and cannot do [6]. For instance, they are asked to modify the numbers in one of the glide blocks in Figure 7 and describe how the glide block functions and where it can be used.

In the second investigation, students are given a “messy” version of the water molecule cycle and they are asked to modify the code in order to debug it. Thus, they engage in an iterative process of predicting an outcome, acting it out, and modifying their prediction accordingly until they are satisfied with the result [12].

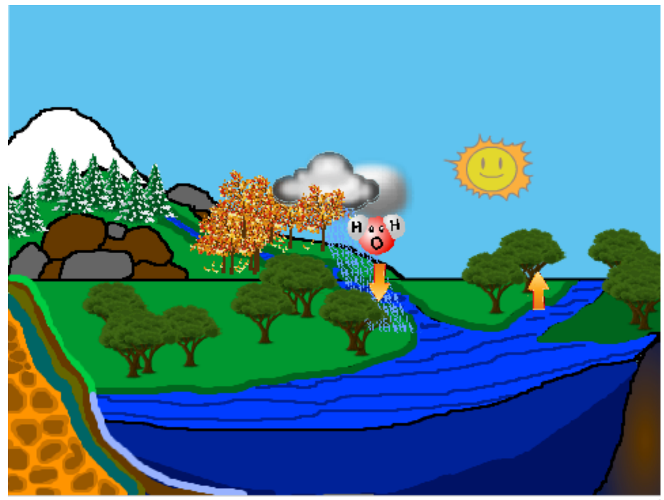

FIGURE 5
FIRST LOOP OF THE WATER MOLECULE

FIGURE 6
SECOND LOOP OF THE WATER MOLECULE

By discussing how the different components of the simulation work, how to modify existing code and how to debug and test, students engage in computational practices [2]. Additionally, students engage in computational perspectives [2] by expressing how they understand the water cycle through their design using computational technology with coding and visualization.

FIGURE 7
THE BLOCKS USED FOR PROGRAMMING THE CYCLE OF THE WATER MOLECULE

In terms of mathematical thinking, students explore the ideas of a variable, patterns, coordinate plane, and translations. More specifically, they examine the concept of a variable as a quantity that varies and explore patterns and pattern relationships in the programming language. They become familiar with the coordinate plane of these environments and use geometric transformation reasoning to translate their *Scratch* constructions.

CONCLUSIONS

This paper describes our work in progress for designing an integrated curriculum for EES, computational and mathematical thinking for grade 5-7 students. Currently, we are in the process of designing the modules. After the modules are designed, they will go over an iterative process of refinement with students. We will test the modules with students, monitor students' reasoning and modify the tasks and tools accordingly. We consider our module design to be very important for initiating a discussion around developing an integrated STEM + Computing curriculum and how it can be used for developing a conceptual understanding of teaching and learning in K-12 sciences.

ACKNOWLEDGMENT

This research is supported through a STEM+Computing grant from the Division of Research on Learning of the National Science Foundation (# 1742125).

REFERENCES

[1] Bostrom, A., Morgan, M. G., Fischhoff, B., & Read, D. (1994). What do people know about global climate change? 1. Mental models. *Risk Analysis,* 14(6), 959-970.

[2] Brennan, K., & Resnick, M. (2012). Using artifact-based interviews to study the development of computational thinking in interactive media design. Paper presented at *Annual American Educational Research Association meeting (AERA)*, Vancouver, BC, Canada.

[3] Ainley, J. Pratt, D. & Hansen, A. (2006) Connecting engagement and focus in pedagogic task design. *British Educational Research Journal*, 32(1), 23-38.

[4] Dewey, J. (1938). The philosophy of the arts. *John Dewey: The Later Works*, 13, 357-368.

[5] Kafai, Y. B., & Resnick, M. (1996). *Constructionism in practice: Designing, thinking, and learning in a digital world.* Hillsdale, NJ: Lawrence Erlbaum Associates.

[6] Papert, S. (1980). *Mindstorms: Children, computers, and powerful ideas*. Basic Books, Inc.

[7] Piaget, J., & Duckworth, E. (1970). Genetic epistemology. *American Behavioral Scientist,* 13(3), 459-480.

[8] Wilensky, U. (1999). *NetLogo.* (http://ccl.northwestern.edu/netlogo/). Center for Connected Learning and Computer-Based Modeling, Northwestern University, IL.

[9] Linacre E. (1977) A simple formula for estimating evaporation rates in various climates, using temperature data alone. Agricultural Meteorology, 18(6), 409-424.

[10] Wilensky, U., & Reisman, K. (2006). Thinking like a wolf, a sheep, or a firefly: Learning biology through constructing and testing computational theories—An embodied modeling approach. Cognition and Instruction, 24(2), 171–209.

[11] Maloney, J., Resnick, M., Rusk, N., Silverman, B., and Eastmond, E. (2010). The scratch programming language and environment. ACM Transactions on Computing Education, *10(4)*, Article 16.

[12] Benton, L, Hoyles, C, Kalas, I and Noss, R. (2017). Bridging Primary Programming and Mathematics: some findings of design research in England. Digital Experiences in Mathematics Education, 3(2), pp. 115-138

AUTHOR INFORMATION

Michelle Zhu, Associate Professor, Department of Computer Science, Montclair State University.

Nicole Panorkou, Assistant Professor, Department of Mathematical Sciences, Montclair State University.

Pankaj Lal, Associate Professor, Department of Earth and Environmental Studies & Associate Director, PSEG Institute for Sustainability Studies, Montclair State University.

Sowmith Etikyala, Research Assistant, Department of Computer Science, Montclair State University.

Erell Germia, Research Assistant, Department of Mathematical Sciences, Montclair State University.

Pricilla Iranah, Research Assistant, Department of Earth and Environmental Studies, Montclair State University.

Bharath Samanthula, Assistant Professor, Department of Computer Science, Montclair State University.

Debasmita Basu, Research Assistant, Department of Mathematical Sciences, Montclair State University.