Infusing Computational Thinking into Science Education

Irene Lee
Paige Prescott
Maureen Psaila-Dombrowski
September 21, 2015

With support from the National Science Foundation
DRL 0639637, OCI 1057672, and CNS 1240992.
Outline of today’s presentation

• Background
  ▫ Complexity Science and Science in the 21\textsuperscript{st} Century
  ▫ Computational thinking
  ▫ NRC Framework’s “Scientific Practice” dimension

• Programs that integrate CT and Science via Modeling & Simulation

• Preparing teachers as computational thinkers

• Questions and answers
Complexity Science at Santa Fe Institute
Current Research Strengths

**Cities**
- Theory of cities
- Scaling relationships
- Informal settlements

**Networks**
- Community structure
- Dynamics
- Statistical inference
- Ecological
- Epidemiological
- Social
- Technological

**Cultural Evolution**
- Emergence of states
- Wealth inequality
- Cooperation
- Social networks
- Human behavior

**Biological Evolution**
- Origins of life/metabolism
- Major transitions
- Multicellularity
- Molecular evolution
- Regulatory systems
- Diversity

**Information Theory**
- Information processing
- Biological/social complexity
- Mutual info. & evolution
- Cognition & computation

**Computation**
- Quantum computing
- Optimization
- Phase transitions
- Machine learning
- Big data issues

**Innovation**
- Technological
- Biological
- Conceptual

**Sustainability**
- Theory of?
- Urbanization
- Human ecology
- Ecological dynamics
- Robustness/Resilience/Risk

**Linguistics**
- Evolution of language
- Language phylogenies
- Cultural shifts

**Economics**
- Financial Markets
- Risk/Debt/Leverage
- Tech Innovation
Current Research Strengths

Cities
- Theory of cities
- Scaling relationships
- Informal settlements

Networks
- Community structure
- Dynamics
- Statistical inference
- Ecological
- Epidemiological
- Social
- Technological

Cultural Evolution
- Emergence of states
- Wealth inequality
- Cooperation
- Social networks
- Human behavior

Biological Evolution
- Origins of life/metabolism
- Major transitions
- Multicellularity
- Molecular evolution
- Regulatory systems
- Diversity

Innovation
- Technological
- Biological
- Conceptual

Sustainability
- Theory of?
- Urbanization
- Human ecology
- Ecological dynamics
- Robustness/Resilience/Risk

Linguistics
- Evolution of language
- Language phylogenies
- Cultural shifts

Information Theory
- Information processing
- Biological/social complexity
- Mutual info & evolution
- Cognition & computation

Computation
- Quantum computing
- Optimization
- Phase transitions
- Machine learning
-_big data issues

Economics
- Financial Markets
- Risk/Debt/Leverage
- Tech Innovation
Characteristics of Complex Systems

- Made up of many parts or agents
- Agents follow simple rules
- Interactions - agents and environment
- Emergent pattern arise from interactions
- Leaderless
- Hard to predict
- Bottom up rather than top down
Science in the 21st Century

Urgent need to understand large complex systems to address the problems of the 21st century that affect us all such as climate change, loss of biodiversity, energy consumption and virulent disease.
Computational Science
Increases in computational power enable us to:

• design and conduct experiments on models of systems too big, too expensive or too dangerous to experiment with in the real-world.

• run multiple “what-if” scenarios quickly.

• collect and analyze large amounts of data.
Computational Scientist’s toolkit

New Fields:
- Computational Biology
- Computational Physics
- Computational Social Science
- Computational Chemistry

Computer Modeling and Simulation:
- Agent based modeling
- Stochastic modeling
- Monte Carlo simulation
- Systems dynamics modeling
Computational Thinking

Computer Modeling and Simulation:
- Agent based modeling
- Stochastic modeling
- Monte Carlo simulation
- Systems dynamics modeling

Computational Thinking:
- Abstraction
- Automation
- Analysis
## Computational Thinking in K-12

<table>
<thead>
<tr>
<th>Activity</th>
<th>Abstraction</th>
<th>Automation</th>
<th>Analysis</th>
</tr>
</thead>
<tbody>
<tr>
<td>Modeling &amp; Simulation</td>
<td>Selecting features of real-world to incorporate in a model.</td>
<td>Time stepping using a model as an experimental testbed.</td>
<td>Were the correct abstractions made? Does the model reflect reality?</td>
</tr>
<tr>
<td>Robotics</td>
<td>Design robot to react to a set of conditions.</td>
<td>Program checks sensors to monitor conditions.</td>
<td>Are there situations that were not taken into account?</td>
</tr>
<tr>
<td>Game Design &amp; Development</td>
<td>Games are abstracted into a set of scenes containing characters</td>
<td>Game responds to user actions.</td>
<td>Do the elements incorporated make the game fun to play?</td>
</tr>
</tbody>
</table>
Next Generation Science Standards

NRC Framework:
“Vision” document

Standards are Performance Expectations (PEs):

Every standard has three dimensions:
1. Science/Engineering practices (SEPs) = practice
2. Disciplinary core ideas (DCIs) = content
3. Cross-cutting concepts (CCs) = themes
Science and Engineering Practices

1. Asking questions and defining problems
2. Developing and using models
3. Planning and carrying out investigations
4. Analyzing and interpreting data
5. Using mathematics and computational thinking
6. Constructing explanations and designing solutions
7. Engaging in argument from evidence
8. Obtaining, evaluating, and communicating information
Developing and using models in the NGSS Performance Expectations

5-LS2-1. Develop a model to describe the movement of matter among plants, animals, decomposers, and the environment.

MS-PS1-4. Develop a model that predicts and describes changes in particle motion, temperature, and state of a pure substance when thermal energy is added or removed.

MS-LS3-2. Develop and use a model to describe why asexual reproduction results in offspring with identical genetic information and sexual reproduction results in offspring with genetic variation.

MS-ETS1-4. Develop a model to generate data for iterative testing and modification of a proposed object, tool, or process such that an optimal design can be achieved.

HS-PS1-4. Develop a model to illustrate that the release or absorption of energy from a chemical reaction system depends upon the changes in total bond energy.
Create, revise, and use computational models in the NGSS Performance Expectations

<table>
<thead>
<tr>
<th>Code</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>HS-PS3-1</td>
<td>Create a computational model to calculate the change in the energy of one component in a system when the change in energy of the other component(s) and energy flows in and out of the system are known.</td>
</tr>
<tr>
<td>HS-LS2-1</td>
<td>Use mathematical and/or computational representations to support explanations of factors that affect carrying capacity of ecosystems at different scales.</td>
</tr>
<tr>
<td>HS-LS4-6</td>
<td>Create or revise a simulation to test a solution to mitigate adverse impacts of human activity on biodiversity.*</td>
</tr>
<tr>
<td>HS-ESS3-3</td>
<td>Create a computational simulation to illustrate the relationships among management of natural resources, the sustainability of human populations, and biodiversity.</td>
</tr>
<tr>
<td>HS-ESS3-6</td>
<td>Use a computational representation to illustrate the relationships among Earth systems and how those relationships are being modified due to human activity.</td>
</tr>
<tr>
<td>HS-ETS1-4</td>
<td>Use a computer simulation to model the impact of proposed solutions to a complex real-world problem with numerous criteria and constraints on interactions within and between systems relevant to the problem.</td>
</tr>
</tbody>
</table>
Other examples in the NGSS (where CT and Modeling and Simulation are applicable)

**MS-LS2-4**  
Construct an argument supported by empirical evidence that changes to physical or biological components of an ecosystem affect populations.

**MS-ESS3-5**  
Ask questions to clarify evidence of the factors that have caused the rise in global temperatures over the past century.

**MS-PS1-5**  
Apply scientific principles and evidence to provide an explanation about the effects of changing the temperature or concentration of the reacting particles on the rate at which a reaction occurs;

**HS-ETS1-1**  
Analyze a major global challenge to specify qualitative and quantitative criteria and constraints for solutions that account for societal needs and wants.
Our Approach: The Computational Science Cycle

1. **Real World Problem**
2. **Simplify**
3. **Translate**
4. **Simulate**
5. **Parameterize**
6. **Experimental Testbed**
7. **Results / Conclusions**
8. **Interpret**
9. **Idea for a Model**
10. **Computational Model**
Our Approach:
The Computational Science Cycle

Analysis

Automation

Abstraction

Automation

Abstraction

Automation
Our Approach: The Computational Science Cycle

1. Real World Problem
2. Idea for a Model
3. Experimental Testbed
4. Computational Model
5. Results / Conclusions
6. Simulate
7. Interpret
8. Translate
9. Parameterize
10. Simplify
11. Abstraction

Analysis

Automation

Automation

Abstraction
Our Approach: The Computational Science Cycle

- Real World Problem
- Simplify
- Idea for a Model
- Translate
- Computational Model
- Parameterize
- Experimental Testbed
- Simulate
- Interpret
- Results / Conclusions
- Analysis
- Abstraction
- Automation
Our Approach: The Computational Science Cycle
The Computational Science Cycle and Scientific and Engineering Practices

1. Asking questions / defining problems
2. Developing and using models
3. Planning and carrying out investigations
4. Analyzing and interpreting data
5. Using math and computational thinking
The Computational Science Cycle and Scientific and Engineering Practices

1. Asking questions / defining problems
2. Developing and using models
3. Planning and carrying out investigations
4. Analyzing and interpreting data
5. Using math and computational thinking
6. Constructing explanations
7. Engaging in argument from evidence
8. Obtaining, evaluating, and communicating info.
The Computational Science Cycle and Scientific and Engineering Practices

1. Asking questions / defining problems
2. Developing and using models
3. Planning and carrying out investigations
4. Analyzing and interpreting data
5. Using math and computational thinking
6. Constructing explanations
7. Engaging in argument from evidence
8. Obtaining, evaluating, and communicating info.

Compare outcomes with what is known about the real world—to see if they “make sense.”
An example:

Compare generated data to real-world data

Run simulations and collect data

computer model
Two SFI programs that Integrate Computational Thinking and Science.
Project GUTS offers:

Afterschool clubs for middle school students in NM
Fieldtrips/Conferences and Student Roundtables

Regular school day integration into MS Science classes in 18 districts across the US through Code.org.
Project GUTS Framework:

- Use-Modify-Create trajectory
- Studying phenomena as complex systems
  ex) Spread of disease, traffic patterns, pollution, ecosystems,
      emergency egress, shared resources and sustainability,
      social networks, and opinion dynamics
- Place-based education (in the afterschool programs)
- Near-peer mentoring
- Meeting developmentally appropriate needs:
  comfortable context, program flexibility, and social
  environment.
Supporting Growth in Computational Thinking

• Provide opportunities to learn and practice basic CS concepts and constructs
Provide opportunities to think about models

- Comparing and contrasting real world “participatory simulations” and the same activities within a computer model.

Setup in a large circle (at least 4 ft away from your neighbors)

Then, at my command, follow these two instructions:

1. Turn to the person on your left and set that direction as your heading;

2. Take 3 steps forward in that direction.
Provide opportunities to think about models

• Comparing and contrasting real world “participatory simulations” and the same activities within a computer model.
Supporting Growth in Computational Thinking

- Rich computational environments allow users to “look under the hood” and see how it works.
Use-Modify-Create progression

- USE
- MODIFY
- CREATE

“Not Mine”

“Mine”

Refine → Test → Analyze

Use-Modify-Create progression diagram
Modeling and Simulation

- **Agent based modeling**
  - StarLogo TNG (*education.mit.edu/starlogo-tng*)
  - StarLogo Nova (*slnova.org*)
  - NetLogo (*ccl.northwestern.edu/netlogo*)
Progression of skill building

Agent creation

Agent movement

Agent interaction with environment

Agent interaction with other agents

First “scientific” model - contagion
Integrating “CS in Science”

• 4 curricular modules by Project GUTS for Code.org
  ▫ Introduction to Modeling and Simulation
  ▫ Earth Science – Water resources
  ▫ Life Science – Ecosystems as complex systems
  ▫ Physical Science – Chemical reactions

• Each is a 5 hour “replacement” unit that follows the Use-Modify-Create trajectory
Scaffolding learning

• Model observation form
  (analyze and decode a model)

• Experimental design form
  (use model as experimental test bed)

• Model design form
  (make modifications to base model)

• Project design form
  (create your own model and experiments from scratch.)
Agency: the capacity to act, create and take ownership

- Students are engaged in project-based learning.
- Students act as creators, innovators, scientists.
- Ownership of project, investigation, creation.
- Interaction between students (collaboration).
- Iterative refinement of projects (persistence).
- Capability to take ownership, make change in the world.
Resources:

- Project GUTS website (projectguts.org)
- Project GUTS / Code.org CS in Science (available at code.org/curriculum/science)
- Project GUTS online PD course / MOOC (available at guts-cs4hs.appspot.com)
New Mexico Computer Science for All

• A comprehensive teacher professional development program in Computer Science.
What is NM-CSforAll?

The comprehensive year-long teacher professional development program in Computer Science offers:

- Professional Development for teachers via a UNM Computer Science course offered online.
- Quarterly workshops: face-to-face and online.
- Summer PD focused on pedagogy
- Practicum: Computer Science course offered for high school students.
NM-CSforAll’s goals

• The program’s goals are to:
  • Prepare high school STEM teachers to be future CS teachers.
  • Create pathways for implementation of CS classes within NM high schools.
  • Increase the number and diversity of students taking CS classes in NM high schools.
About the program:

• Features the Big Ideas of CS from AP “CS Principles” framework.
• Uses Modeling & Simulation as an introduction to Computer Science
• Demonstrates the wide applicability of CS
Example: Ecosystems

• Ecosystem model - built in stages
  • Where is Nemo?
    • Agents interacting with environment (patches)
    • Fish eating plankton
  • Saving Nemo
    • Introduces birth, death and population dynamics
  • Eating Nemo
    • Agents interacting with other agents
    • Predator /Prey
Eating Nemo
Example: Ecosystems

• How students extended the Ecosystems model:
  • Ecosystem with an infections
  • Effect of Hunting
  • Competition for food source
  • Multilevel food chains
Potential to impact STEM education

- Teachers can develop modules for STEM content areas
- Students can apply modeling and simulation techniques
- Projects for competition: Supercomputing Challenge
  - Using a Concentrated Heat System to Shock the P53 Protein to Direct Cancer Cells into Apoptosis
  - Factors Affecting Solar Power Production Efficiency
  - The Effects of Motivation for a Healthy Lifestyle
  - Factors Affecting The Rate Of Co2 Emissions From Vehicles Of Different Engine Types
Preparing CT-enabled teachers

• Take the graduate level UNM CS590 to learn CS concepts, skills and pedagogy.
• Summer Bootcamp to focus on pedagogy
• Serve as learning coach during the lab portion of Fall 2014 dual credit UNM CS108L (offered as a regular school day class within high schools.
• Learn and use pedagogy and best practices for engaging and retaining underrepresented students in computing.
Preparing CT-enabled teachers

• Build confidence with Computer Science concepts and constructs
• Experience
  ▫ Computational modelers
  ▫ Developing and using models as test beds.
  ▫ Developing inquiry-based projects.
• Pedagogy
Resources for educators:

- Graduate Level class (UNM CS590)
- MOOC Starting November 2, 2015
- Open curriculum on CS10K website
  - [https://cs10kcommunity.org/resource/3523](https://cs10kcommunity.org/resource/3523)
- Teachers who complete UNM CS590 have access to a teacher’s site with extensive curriculum.
Questions?

Contact information:

Irene Lee
lee@santafe.edu

Maureen Psaila-Dombrowski
mpd@santafe.edu

Paige Prescott
paige@santafe.edu
THE END
Preparing CT-enabled teachers

- Project GUTS Professional development
- Build experience as computational thinkers
- Gain comfort with CS concepts and constructs
- Learn pedagogy
Code.org’s Middle School CS in Science Professional development program

- Spring: Pre-workshop online prep (~2 hours)
- Summer: In-person workshop (3 days, ~24 hours)
- Summer: Online reflection and planning (~8 hours)
- School Year: Online and In-person follow up

Workshop costs and teacher stipends covered by Code.org