Computational Thinking Without Writing Code: What's Next for Computational Modeling?

Tamar Fuhrmann (chair), TC Columbia University, tamarrf@gmail.com
Gabrielle Rabinowitz (chair), American Museum of Natural History (AMNH), grabinowitz@amnh.org
Amanda Peel, New Mexico State University, apeel@nmsu.edu
Sharin Jacob, Digital Promise Global, sjacob@digitalpromise.org
Quinn Burke, Digital Promise Global, qburke@digitalpromise.org
Ari Krakowski, University of California Berkeley's Lawrence Hall of Science, akrakowski@berkeley.edu
Eric Greenwald, University of California Berkeley's Lawrence Hall of Science, eric.greenwald@berkeley.edu
Preeti Gupta, AMNH, pgupta@amnh.org
Rachel Chaffee, AMNH, rchaffee@amnh.org
Emma Anderson, MIT, eanderso@mit.edu
Kirsten Peterson, Education Development Center, Inc., kpeterson@edc.org
Joyce Malyn-Smith, Education Development Center, Inc. jmsmith@edc.org
Beatriz Perret, Boston College, perretg@bc.edu
John Chen, Northwestern University, yuehanchen2023@u.northwestern.edu
Uri Wilensky, Northwestern University, rchaffee@amnh.org
Daniel Wendel, MIT Scheller Teacher Education Program, starlogodaniel@gmail.com
Aileen Han, MIT Scheller Teacher Education Program, ahan@mit.edu
Aditi Wagh, MIT Scheller Teacher Education Program, awagh@education.mit.edu
Paulo Blikstein, Columbia University, USA, and I. Tecnológico de Monterrey, Mexico, paulob@tc.columbia.edu
Michelle Wilkerson, University of California Berkeley, mwilkers@berkeley.edu
Kate Moore, MIT, ksmoore@education.mit.edu
Helen Zhang, Boston College, zhangzm@bc.edu
Mark Weckel, AMNH, mweckel@amnh.org
Sheikh Ahmad Shah, Boston College, shahst@bc.edu
Ji Hye Park, Boston College, jihye.park@bc.edu
Safinah Ali, New York University, safinahali@nyu.edu
Alex Perez, AMNH, aperez@amnh.org

Shuchi Grover (discussant), Looking Glass Ventures, shuchi.stanford@gmail.com
Irene Lee (discussant), MIT, ireneaelee@gmail.com

Abstract: Generative Artificial Intelligence (AI) introduces exciting new possibilities and challenges to the established field of computational modeling education. The ten posters in this symposium provide different perspectives on the changing landscape and present examples of educational programs, professional development strategies, pedagogical approaches, and digital tools that have helped learners and educators develop the skills needed to interrogate and co-create scientific computational models with AI. The posters address each stage of computational modeling education, including reading and decoding models, modifying existing models, creating (or co-creating) models, and evaluating models.

Introduction

Integrating computational thinking (CT) into science education equips students with powerful tools for exploring scientific phenomena, fostering deeper conceptual understanding, and engaging in authentic scientific practices. The interdisciplinary potential of this work empowers students and practitioners to engage meaningfully with emerging local and global challenges (Barr et al., 2011; Weintrop et al., 2016). Existing literature identifies several core CT practices: abstraction, generalization, decomposition, algorithmic thinking, and debugging (Caeli & Yadav, 2020). Computational modeling, including agent-based modeling (ABM), has long been seen as a valuable tool that can simultaneously support students in science learning and CT (e.g., Wilkerson et al., 2015). Recently, however, developments in Artificial Intelligence (AI) have introduced new possibilities for how learners and educators may interact with computational models (Chen et al., 2023). In the future, it may no longer be necessary to write code to create scientific computer models, putting learners in the role of collaborators with generative AI. As a result, increasing emphasis must be placed on skills such as "code sense," the intuitive ability to read and understand code without necessarily writing it (Johnson, 2024), and "decoding," mapping mechanisms in code onto scientific processes (Lee, 2024).

This timely symposium presents research findings across ten projects, which inform the tools, practices, and approaches needed to ensure students develop the skills needed to interrogate and co-create models. We aim to contribute to the ongoing conversation about what it means to model scientific phenomena in a world where AI plays a central role in computational work. In this symposium, we present posters that reimagine computational modeling education at every stage: 1) reading and decoding models, 2) modifying existing models, 3) creating (or co-creating) models, and 4) evaluating and validating models. Peel et al. (*Poster #1*) present evidence suggesting that learners benefit from an introduction to unplugged algorithm concepts before engaging with computational models. Jacob et al. *(Poster #2)* explore how scaffolding strategies from literacy studies can enhance learners' ability to navigate computational tasks while reading and decoding models. Krakowski et al. (*Poster #3*) illustrate instructional strategies for developing and evaluating computational models in science. Rabinowitz et al. *(Poster #4)* highlight the role of decoding skills in supporting students taking on the role of model "analyst." Anderson et al. (*Poster #5*) explore the implementation of the decoding approach across three projects with 6th–12th grade students both in classrooms and in out-of-school programs. Peterson et al. (*Poster #6*) emphasize the importance of the use-decode-modify model to enhance science understanding through computational thinking. Chen et al. (*Poster #7*) and Wendel et al. (*Poster #8*) introduce the unique opportunities and challenges of co-creating computational models with generative AI. Fuhrmann et al. *(Poster #9)* present strategies to support students in validating computational models by comparing simulation results to real-world data. Finally, Moore et al. (*Poster #10)* consider what support students and educators need in fundamental AI concepts (AI literacy) to engage meaningfully in the future of computational modeling.

Following an introduction to the symposium by the session chairs, each research team will have two minutes to introduce their research and key findings briefly. This will be followed by open circulation among the ten posters. Finally, our discussants, Dr. Shuchi Grover and Irene Lee, will draw upon the various perspectives and summarize the insights from everyone's work, followed by discussion and Q&A.

Poster 1: Pre-service science teacher learning about computational thinking through computational modeling practices

Amanda Peel

With the introduction of AI, the need to code from scratch is diminishing because AI will and is beginning to be able to generate code for users. It is now imperative that people know how to read, understand, debug, and modify code to fit their needs. As such, this work focuses on building foundational computational thinking (CT) skills by introducing learners to unplugged, or computer-free, algorithm concepts and using it as a scientific practice. In this approach, learners are introduced to algorithm concepts (iteration, branching, methods, iteration) and use them to create hand-written algorithms that explain a science phenomenon (Peel, Sadler, & Friedrichsen, 2022). Next, learners extend their CT practice by building and modifying science computational models with blocks. This allows them to engage in computational modeling without having to know a programming language or syntax and allows learners to learn to use the algorithm concepts in a computational programming environment, which fosters building and debugging code. As learners build their model, they can compare their block code to their algorithmic explanations, fostering the deepening of CT practices. Next, learners can look at the NetLogo code associated with the block program they made. Aligning the code to the block allows learners to begin to understand how to read code because they can see what pieces of code are added when they add a block (i.e., they know what the block does in the model, so they know what that piece of code does now, too). The work shared in this session is from a science teacher education course where learners are pre-service (PSTs) or alternative licensure teachers (ALTs). A series of lessons were designed following the steps described above. These lessons were piloted and data was collected to answer the following research question: How did PSTs and ALTs learn about CT through their engagement in the CT lessons? Findings will be presented and discussed in the session.

Poster 2: Integrating literacy-based scaffolding to support decoding of computational models

Sharin Jacob, Quinn Burke

With the rise of generative artificial intelligence, which produces code syntax outputs based on programming queries, learning to read and understand code has become more important than ever. Computational modeling involves the use of computer-based simulations to represent real-world systems. Understanding computational models requires students to decode complex systems of symbols, structures, and relationships—a process similar to reading and interpreting text (Vogel et al., 2020). This proposal explores how scaffolding strategies from literacy research can be adapted to help students designated as English learners (ELs) develop the skills to decode computational models. Scaffolding, a core component of literacy instruction, involves breaking down complex tasks into manageable steps while building on students' prior knowledge (Clark & Graves, 2005). For example, guided reading strategies such as examining textual features, making predictions, and summarizing can be adapted to computational contexts (Avalos et al., 2024; Salac et al., 2020). When applied to decoding computational models, these strategies include breaking down code into logical segments, predicting model outcomes, and summarizing the purpose of a model.

Scaffolding frameworks are most successful when combined with culturally responsive and empirically supported practices to support students designated as ELs. This includes leveraging students' home languages as assets (Prado et al., 2022), providing multimodal instruction (NASEM, 2018), incorporating code translations (Vogel et al., 2020), and including collaborative, discourse-rich opportunities for sense-making about CT (Nguyen et al., 2020). Such approaches ensure that students build both the conceptual understanding and the disciplinary language needed to engage in CT activities.

In our poster, we will provide a research-based strategy for scaffolding both reading and coding through the structured decoding of computational models. By explicitly connecting decoding in literacy research to the decoding of computational models, we seek to help students to navigate complex systems while fostering their interdisciplinary learning.

Poster 3: Unlocking scientific simulations: Strategies for computational sense-making in science

Ari Krakowski, Eric Greenwald

The growing centrality of computation in contemporary science practice has prompted calls to integrate computational thinking (CT) within science education in an effort to more authentically align science learning with disciplinary practice and more capably prepare students for participation in STEM fields (Hurt et al., 2023; Lee & Malyn-Smith, 2020). Our approach to CT integration is grounded in a stance toward CT from the *perspective of science* and from the *perspective of the learner. It offers* an alternative for classroom science instruction that is a better bridge both to the ways science is currently practiced AND to the applications of science that resonate with youth. Core to this approach is asset-based learning experiences grounded in computational science inquiry that are authentic to the increasingly computational nature of science as a modern practice as well as responsive to the lived experiences of youth.

For this symposium, we will present instructional strategies for developing and evaluating computational models in science. The strategies have been empirically tested to reveal statistically significant learning gains and increased competency beliefs for CT, with particularly pronounced gains among female-identifying and BIPOC students (Krakowski et al., 2024). The pedagogical approaches help students develop competence inspecting and thoughtfully using computational tools both as computational artifacts (understanding how the tool is designed to function) and as tools for science (understanding how the tool can serve to answer a science question). We also explore emerging ways that AI, as a computational tool, is transforming how scientists and learners can engage with and expand capacities for agent-based modeling. Strategies include structured reflection and sensemaking activities about computational models, non-digital/no-code activities that work toward algorithmic explanations (Peel et al., 2022) of model behavior, and structured student discourse aimed at deepening understanding of scientific simulations as computational tools.

Poster 4: The "decoding-first" approach: A shift from learners as model creators to model analysts

Gabrielle Rabinowitz, Irene Lee, Preeti Gupta, Rachel Chaffee

As we prepare youth to engage with the complex phenomena of a changing world, there has been a shift away from seeing science learning as mere retention of science facts towards understanding how and why phenomena occur (NGSS Lead States, 2013). However, complex systems are difficult for middle school students to grasp (Gotwals & Songer, 2010; Riess & Mischo, 2010). Agent-based models provide an excellent opportunity for students to apply mechanistic reasoning and computational thinking practices as they progress through a "Use-Modify-Create" (UMC) trajectory (Lee et al., 2011). With the advent of generative AI, the process of generating agent-based models may be increasingly automated (Chen et al., 2024). In contrast to the "programming-first" approach to the integration of CT in science, "Decoding" involves mapping between processes in science and mechanisms in code, which the learner may or may not have written themselves (Lee, 2024). We propose that the decoding-first approach not only supports students in understanding complex scientific phenomena but also provides them with the skills necessary to critique and analyze agent-based models of any origin.

In this poster, we present the findings of a mixed-methods study of the DecodeNYC summer program offered to New York City 7th and 8th graders at the American Museum of Natural History (AMNH) in 2020 - 2022, in which participants decode agent-based models of ecosystem dynamics and disease transmission. Across three years of remote, hybrid, and in-person instruction, we observed that decoding existing models and imagined models increased students' capacity for mechanistic reasoning, computational thinking, and science learning. As generative AI continues to grow in capability and influence, youth engaging with agent-based models will increasingly take on the role of analyst and collaborator, making decoding a crucial skill for young learners.

Poster 5: Decoding computational models to support computational thinking for 6th- 12th grade students in a generative AI world

Emma Anderson, Beatriz Perret, Gabrielle Rabinowitz, Irene Lee

The proliferation of generative AI (genAI) tools is reshaping the computational learning landscape, prompting educators to reconsider what is essential for students to understand within computer science (CS) and computational thinking (CT). With genAI able to easily produce workable code and even run simulations, the question arises, whats skills and knowledge should students be learning in K-12 CS/CT education? While there has been a push for increasing access to CS/CT education for a long time now (Proctor, Bigman, & Blikstein, 2019; Wilson, Sudol, & Stephenson, 2010) with many advancing the notion of integrating CS /CT into everyday required subjects within K-12 to support student learning (e.g., Grover & Pea, 2013) education, yet teachers struggle to implement such curriculum due the interplay between their subject and computation (Rich, Yadav & Schwarz, 2019), time constraints, (Kite & Park, 2020) among others. Given this landscape of the changing skills to be proficient in CS/CT, yet the continued need for CS /CT education as understanding how the digital world is built and designed is still of value, we present in this paper a different approach to integration through Decoding (Lee, 2024). Decoding emphasizes creating explicit connections between coded mechanisms and the real-world concepts being modeled. Decoding allows students to engage in computational thinking without a code-first approach.

This paper explores the implementation of the Decoding approach across three STEM+C projects with 6th–12th grade students both in classrooms and in out-of-school programs. Looking across these projects, we explore the approach each project took to integrate decoding. We share initial evidence of decoding supporting students' learning and discuss how this approach may be useful in a world where genAI produces code that software engineers need to evaluate and implement.

Poster 6: Science+C: Integrating a new computational modeling curriculum into high school science classrooms

Joyce Malyn-Smith, Irene Lee, Kirsten Peterson, Beatriz Perret

As computational thinking (CT) and modeling become increasingly central to modern scientific practices, integrating these skills into high school science education is crucial (Weintrop et al., 2016). Recognizing this, the Education Development Center (EDC), in partnership with the Massachusetts Department of Elementary and Secondary Education (MA DESE) and funded by the National Science Foundation (NSF), launched Science+C, a novel computational modeling curriculum that embeds CT within foundational high school science courses.

Unlike traditional programming-centric curricula, Science+C employs a structured "Use, Decode, Modify" framework. Students begin by interacting with existing agent-based models to investigate scientific phenomena such as epidemics and natural selection (Peel, Sadler & Friedrichsen, 2019). They then decode these computational models, analyzing underlying code structures to identify scientific mechanisms at work. Finally, students modify model parameters to pose new scientific questions and test hypotheses. This iterative process aligns closely with authentic scientific inquiry and offers accessible entry points for students regardless of prior coding experience.

Our mixed-methods study conducted during the 2021-2023 academic years in Massachusetts high schools reveals significant benefits of this decoding-focused approach. Teachers reported increased student engagement, improved understanding of scientific concepts, and enhanced computational thinking skills. Additionally, teacher professional development supported successful classroom implementation, creating a sustainable model for integrating CT into high school science. As generative AI transforms computational practices, Science+C positions students not merely as coders, but as critical consumers and analysts of computational models. This approach democratizes access to essential scientific practices, bridging equity gaps and preparing students to participate confidently in an AI-driven future (Margolis et al., 2012).

Poster 7: Exploring the expert-novice gaps in AI-assisted agent-based modeling

John Chen, Uri Wilensky

Our study interrogates the informal, professional learners' perceptions, experiences, and needs of agent-based modeling (ABM) interfaces in the Generative AI (GAI) era. ABM, a critical computational methodology for understanding and studying complex systems (Wilensky & Rand, 2015), has traditionally required significant programming expertise to support tinkering and construction of models (Macal & North, 2013). GAI models, capable of reading and writing natural and programming languages, have opened possibilities to support ABM learners and practitioners (Chen & Wilensky, 2023). Yet, it is still unclear how learners would interact with GAI to achieve their goals and where they would require additional assistance beyond a chat-based assistant.

Through an observation-interview study with 30 global professional participants from diverse backgrounds, our work highlights key differences in how experts and novices perceive, interact with, and hope for LLM-driven ABM interfaces. While the two groups had similar expectations, their experiences were significantly different: experts demonstrated a greater ability to apply human judgment and overcome AI shortcomings, while novices faced challenges in evaluating AI outputs and debugging errors. Our study, uncovered a knowledge gap that prevents novices from fully utilizing GAI's potential. The gap further shaped novices' perceptions of LLM-based interfaces and ABM learning, resulting in less inclination for adoption and also providing motivation for engaging with conventional NetLogo learning materials (such as tutorials or videos).

While GAI models are powerful and capable of constructing ABMs, human learners still need considerable meta-level expertise to gain from LLM-based interfaces. Instead of a silver bullet, chat-based assistants may inadvertently increase the gap between learners, leading to unexpected outcomes in educational equality. To help bridge the gap, we suggest the need for LLM-based interfaces that provide personalized guidance based on individual learners' prior knowledge, and deeper integration into modeling environments that enable tinkering with smaller code snippets.

Poster 8: AI as a thought partner: Leveraging AI to open up opportunities to interrogate models and their code

Daniel Wendel, Aileen Han, and Aditi Wagh, MIT Scheller Teacher Education Program

For over two decades, agent-based modeling (ABM) has been lauded as a scientific and computational practice to model using and reasoning with code (e.g., Wilensky 1990s). However, the specific form of programming a model has shifted over time. Early ABM tools such as StarLogo (Resnick, 1994; Wilensky, 1995) and NetLogo (Wilensky, 1999) used text-based code. While this opened up unprecedented learning opportunities, creating models involved "friction" (Guzdial, 2019) for learners without prior programming experience. Programming with blocks-based ABM code (e.g., Kahn, 2007; Repenning, 2000; Wendel, 2011; Wilkerson et al., 2015) eliminated some of this friction with its visual interface and drag-and-drop functionality.

The advent of AI has presented yet another opportunity to reimagine the learning opportunities around ABM. We build on work on the NetLogo Chatbot (Chen et al., 2024) to further eliminate "friction" by bringing AI as a partner for interrogating and making sense of models and their underlying code in StarLogo Nova. Our prototype is a GPT-4o-based custom GPT (akin to a RAG system) built on the SLNova "Blockument" (a complete block reference) and a highly-refined beginner tutorial. We prompted the GPT to format responses like the tutorial and to use only blocks found in the Blockument. We tested this on a collection of "troubleshooting challenges" from our teacher training materials, and the custom GPT could identify and explain why the student's model was not working as expected and provide suggestions for 4 out of 5 scenarios, with zero block hallucinations.

Moving forward, we will continue to refine the chatbot and integrate it with StarLogo's platform, similar to Patton et al. (2024), and investigate the platform's potential to support learning with ABMs through a series of user studies. We will share preliminary results at the symposium.

Poster 9: Developing evaluation and critical skills by validating scientific models with empirical data

Tamar Fuhrmann, Aditi Wagh, Paulo Blikstein, Michelle Wilkerson

Data is essential for developing and evaluating scientific models, providing a solid foundation for the process of modeling (Fuhrmann et al., 2018; Schwarz et al., 2009). Research suggests that integrating experimental data with scientific models in educational activities can significantly enhance student learning by fostering both conceptual understanding and modeling and meta-modeling skills (Blikstein et al., 2016; Fuhrmann et al., 2018). Furthermore, coupling experiments with modeling encourages students to concentrate on investigative goals, enabling them to compare and validate their findings through triangulation (Gouvea & Wagh, 2018). Incorporating real-world data into modeling not only enriches the learning experience but also mirrors the authentic practices of scientists. This approach inspires students to investigate the core features of a phenomenon, helping them establish criteria for determining model accuracy (Schwarz, Akcaoglu, Ke, & Zhan, 2013). Similar to how scientists utilize empirical data to validate and adjust their models, students compare their simulations with real-world data to ensure their models accurately represent scientific phenomena. This poster introduces MoDa, a web-based computational modeling platform that integrates model-based and data-based features within a single display, enabling direct comparisons between simulations and real-world data (Fuhrmann et al., 2024; Wagh et al., 2023). We examine how this approach supports 6th-grade students as they engage in model building across different subjects, illustrating how data analysis and model refinement can deepen scientific understanding. Traditionally, constructing agent-based models required coding to simulate complex systems, but with advancements in AI, the need for manual coding is diminishing. This shift shifts the focus from programming to a deeper understanding of model functioning and computational thinking. As AI tools become more sophisticated, it is crucial to emphasize the development of students' critical thinking skills—particularly their ability to evaluate models using empirical data.

Poster 10: Let's teach them how AI works: AI literacy curricular and professional development innovations in high school science and interdisciplinary middle school education

Kate Moore, Irene Lee, Helen Zhang, Mark Weckel, Preeti Gupta, Gabrielle Rabinowitz, Sheikh Ahmad Shah, Beatriz Perret, Ji Hye Park, Safinah Ali, Alex Perez, Rachel Chaffee

We posit that decoding and validating AI models necessitates AI literacy, understanding fundamental AI concepts underlying various AI-enabled tools and associated societal and ethical implications. AI literacy enables students and teachers to apply critical thinking to identify affordances and potential biases and decide whether AI outputs, including AI-generated computer models, are trustworthy and appropriate. Uncritical use of AI-enhanced tools can lead to student profiling (Selwyn, 2019), trust in biased algorithms and misleading information (Buolamwini & Gebru, 2018), and the cultivation of colonialist thinking (Williamson & Eynon, 2020). Some argue that the lack of critical reflection on AI output calls for strengthening ethics-informed AI literacy (Zawacki-Richter et al., 2019; Carvalho et al., 2022). Together, these critiques suggest that education *with* AI-enhanced tools will not foster the competencies needed to overcome these challenges - we need to teach our students and teachers *how* AI works through AI literacy education.

In this poster, we present two projects on AI literacy education, with the goals of inspiring the field of agent-based modeling and providing exemplars on how to integrate AI literacy education into existing approaches. The *Science Research Mentorship Program + Machine Learning (ML)* project found that immersing high school students in a ML-integrated science program in after-school settings closed their gap in AI knowledge and corrected their misconceptions of fundamental AI concepts, which better prepared them to apply ML in scientific research (Rabinowitz et al., *in press*). The *Everyday AI (EdAI) project* provided further evidence that innovative PD programs can prepare and empower teachers to teach AI literacy in their classrooms (Zhang et al., 2024). The two projects show the promise and importance of teaching students how AI works so that they think critically about AI outputs and more effectively engage with AI agent-based modeling tools in the future.

References

Allan, V., Barr, V., Brylow, D., & Hambrusch, S. (2010, March). Computational thinking in high school courses. In *Proceedings of the 41st ACM technical symposium on Computer science education* (pp. 390-391).

Avalos, M. A., Plasencia, A., Chavez, C., & Rascón, J. (2007). Modified guided reading: Gateway to English as a second language and literacy learning. The Reading Teacher, 61(4), 318-329.

Blikstein, P., Fuhrmann, T., & Salehi, S. (2016). Using the bifocal modeling framework to resolve "Discrepant Events" between physical experiments and virtual models in biology. Journal of Science Education and Technology, 25, 513-526.

Buolamwini, J., & Gebru, T. (2018, January). Gender shades: Intersectional accuracy disparities in commercial gender classification. In Conference on fairness, accountability and transparency (pp. 77-91). PMLR.

Carvalho, L., Martinez-Maldonado, R., Tsai, Y. S., Markauskaite, L., & De Laat, M. (2022). How can we design for learning in an AI world? *Computers and Education: Artificial Intelligence, 3*, 100053.

Chen, J., & Wilensky, U. (2023). ChatLogo: A Large Language Model-Driven Hybrid Natural-Programming Language Interface for Agent-based Modeling and Programming. arXiv Preprint arXiv:2308.08102. https://arxiv.org/abs/2308.08102

Chen, J., Lu, X., Du, Y., Rejtig, M., Bagley, R., Horn, M., & Wilensky, U. (2024, May). Learning agent-based modeling with LLM companions: Experiences of novices and experts using ChatGPT & NetLogo chat. In Proceedings of the CHI Conference on Human Factors in Computing Systems (pp. 1-18). https://doi.org/10.48550/arXiv.2401.17163

Clark, K. F., & Graves, M. F. (2005). Scaffolding students' comprehension of text. The Reading Teacher, 58(6), 570-580.

Code.org, CSTA, ECEP Alliance (2023). 2023 State of Computer Science Education. Retrieved from https://advocacy.code.org/stateofcs

Falkner, K., & Falkner, N. J. G. (2012). Supporting and structuring "contributing student pedagogy" in Computer Science curricula. Computer Science Education, 22(4), 413–443. https://doi.org/10.1080/08993408.2012.727713

Fuhrmann, T., Rosenbaum, L., Wagh, A., Eloy, A., Wolf, J., Blikstein, P., & Wilkerson, M. (2024). Right but wrong: How students' mechanistic reasoning and conceptual understandings shift when designing agent-based models using data. Science Education, 1–25. https://onlinelibrary.wiley.com/doi/10.1002/sce.21890

Fuhrmann, T., Schneider, B., & Blikstein, P. (2018). Should students design or interact with models? Using the Bifocal Modeling Framework to investigate model construction in high school science. International Journal of Science Education, 40(8), 867-893.

Gotwals, A. W., & Songer, N. B. (2010). Reasoning up and down a food chain: Using an assessment framework to investigate students' middle knowledge. Science Education, 94(2), 259-281.

Gouvea, J. S. & Wagh, A. (2018). Exploring the Unknown: Supporting Students' Navigation of Scientific Uncertainty With Coupled Methodologies. In Kay, J. and Luckin, R. (Eds.) Rethinking Learning in the Digital Age: Making the Learning Sciences Count, 13th International Conference of the Learning Sciences (ICLS) 2018, Volume 1. London, UK: International Society of the Learning Sciences.

Graebner, C. (2018). How to Relate Models to Reality? An epistemological framework for the validation and verification of computational models. Journal of Artificial Societies and Social Simulation, 21(3), 8.

Grover, S., & Pea, R. (2013). Computational thinking in K–12: A review of the state of the feld. Educational Researcher, 42(1), 38–43.

Guzdial, M. (2019). A new definition of Computational Thinking: It's the Friction that we want to Minimize unless it's Generative. Computing Education Research Blog. https://computinged.wordpress.com/2019/04/29/what-is-computational-thinking-its-the-friction-that-we-want-to-minimize/

Hammond, R. A. (2015). Considerations and Best Practices in Agent-Based Modeling to Inform Policy. In Wallace R, Geller A, Ogawa VA, (Eds.), Assessing the Use of Agent-Based Models for Tobacco Regulation. Washington (DC): National Academies Press (US); Appendix A.

Hurt, T., Greenwald, E., Allan, S., Cannady, M. A., Krakowski, A., Brodsky, L., ... & Dorph, R. (2023). The computational thinking for science (CT-S) framework: Operationalizing CT-S for K–12 science education researchers and educators. International Journal of STEM Education, 10(1), 1.

Jocius, R., O'Byrne, W. I., Albert, J., Joshi, D., Robinson, R., & Andrews, A. (2021). Infusing Computational Thinking into STEM Teaching: From Professional Development to Classroom Practice. Educational Technology & Society, 24(4), 166–179. https://www.jstor.org/stable/48629253

Johnson, M. (2024). Generative AI and CS education. Communications of the ACM, 67(4), 23–24.

https://doi.org/10.1145/3632523

Kahn, K. (2007). Building computer models from small pieces. In G. Wainer (Ed.), *SCSC Proceedings of the 2007 Summer Computer Simulation Conference* (pp. 931–936).

Kelley, T. R., Knowles, J. G., Holland, J. D., & Han, J. (2020). Increasing high school teachers' self-efficacy for integrated STEM instruction through a collaborative community of Practice. International Journal of STEM Education, 7(1). https://doi.org/10.1186/s40594-020-00211-w

Kite, V., & Park, S. (2020). Secondary science teachers' conceptualizations of computational thinking and perceived barriers to CT/content integration. Prepared for the 2020 annual meeting of the National Association for Research in Science Teaching and teacher education (NARST)

Krakowski, A., Greenwald, E., Roman, N., Morales, C., & Loper, S. (2024). Computational Thinking for Science: Positioning coding as a tool for doing science. Journal of Research in Science Teaching, 61 (7), 1574-1608.

Lee, I. (2024). The Decoding Approach to CT Integration. MIT STEP Lab white paper. https://education.mit.edu/uploads/2024/03

Lee, I., & Malyn-Smith, J. (2020). Computational thinking integration patterns along the framework defining computational thinking from a disciplinary perspective. Journal of Science Education and Technology, 29 (1), 9-18.

Macal, C. M., & North, M. J. (2013). Successful approaches for teaching agent-based simulation. Journal of Simulation, 7 (1), 1–11. https://doi.org/10.1057/jos.2012.1

Margolis, J., Ryoo, J. J., Sandoval, C. D. M., Lee, C. H., Goode, J., & Chapman, G. (2012). Beyond access: Broadening participation in high school computer science. *ACM Inroads, 3*(4), 72–78.

National Academies of Sciences, Engineering, and Medicine (NASEM). (2018). English Learners in STEM Subjects: Transforming Classrooms, Schools, and Lives. Retrieved from https://www.nap.edu/catalog/25182/english-learners-in-stem-subjects-transforming-classrooms-schools-and-lives

NGSS Lead States. (2013). Next Generation Science Standards: for states, by states. Washington, DC: The National Academies Press.

Nguyen, H., Garcia, L., Jacob, S., & Warschauer, M. (2020, January). Reflection as formative assessment of computational thinking in elementary grades. In *Proceedings of the International Conference on the Learning Sciences*.

Patton, E. W., & Kim, D. Y. J. et al. (2024). Aptly: Making Mobile Apps from Natural Language. ArXiv. https://doi.org/10.48550/arXiv.2405.00229

Peel, A., Sadler, T. D., & Friedrichsen, P. (2022). Algorithmic explanations: An unplugged instructional approach to integrate science and computational thinking. Journal of Science Education and Technology, 31 (4), 428-441.

Peel, A., Sadler, T. D., & Friedrichsen, P. (2019). Learning natural selection through computational thinking: Unplugged design of algorithmic explanations. *Journal of Research in Science Teaching, 56*(7), 983-1007.

Prado, Y., Ramos, M. N., Peña, E., & Zavala, J. (2022). Dual-language engagement: Concerted cultivation of Spanish use among students, teachers, and parents. Bilingual Research Journal, 45 (2), 159-179.

Proctor, C., Bigman, M., & Blikstein, P. (2019). Defining and designing computer science education in a K12 public school district. In Proceedings of the 50th ACM Technical Symposium on Computer Science Education (pp. 314–320). ACM. https://doi.org/10.1145/3287324.3287440

Rabinowitz, G., Moore, K. S., Ali, S., Weckel, M., Lee, I., Gupta, P., Chaffee R. (2024, *in press*). Results of an Effective Informal Science-integrated High School Machine Learning Curriculum. *International Journal of Science Education.*

Repenning, A. I. (2000). *AgentSheets: End-User Programmable Simulations* [Text.Article]. http://jasss.soc.surrey.ac.uk/3/3/forum/1.html

Resnick, M. (1994). Turtles, Termites and Traffic Jams: Explorations in Massively Parallel Microworlds. MIT Press, Cambridge, MA.

Riess, W., & Mischo, C. (2010). Promoting systems thinking through biology lessons. *International Journal of Science Education, 32(6), 705-725.

Salac, J., Thomas, C., Butler, C., Sanchez, A., & Franklin, D. (2020, February). TIPP&SEE: a learning strategy to guide students through use-modify Scratch activities. In Proceedings of the 51st ACM technical symposium on computer science education (pp. 79-85).

Schwarz, C. V., Akcaoglu, M., Ke, L., & Zhan, L. (2013). Fifth-grade students' engagement in modeling practice across content areas: What epistemologies in practice change over time and how. In Annual meeting.

Selwyn, N. (2019). What's the problem with learning analytics?. *Journal of Learning Analytics, 6*(3), 11-19.
Wagh, A., Fuhrmann, T., Eloy, A. A. da S., Wolf, J., Bumbacher, E., Blikstein, P., & Wilkerson, M. (2022). MoDa: Designing a Tool to Interweave Computational Modeling with Real-world Data Analysis for Science Learning in Middle School. Interaction Design and Children, 206–211. https://doi.org/10.1145/3501712.3529723
Weintrop, D., Beheshti, E., Horn, M. S., Orton, K., Jona, K., Trouille, L., & Wilensky, U. (2016). Defining computational thinking for mathematics and science classrooms. Journal of Science Education and Technology, 25(1), 127-147.
Wilkerson, M., Wagh, A., & Wilensky, U. (2015). Balancing Curricular and Pedagogical Needs in Computational Construction Kits: Lessons From the DeltaTick Project. Science Education, 99(3), 465–499. https://doi.org/10.1002/sce.21157
Williamson, B., & Eynon, R. (2020). Historical threads, missing links, and future directions in AI in education. *Learning, Media and Technology, 45*(3), 223-235. https://doi.org/10.1080/17439884.2020.1798995
Wilson, C., Sudol, L. A., Stephenson, C., & Stehlik, M. (2010). Running on empty: The failure to teach K–12 computer science in the digital age. ACM. https://doi.org/10.1145/3414583
Zhang, H., Lee, I., & Moore, K. (2024, March). An Effectiveness Study of Teacher-Led AI Literacy Curriculum in K-12 Classrooms. In *Proceedings of the AAAI Conference on Artificial Intelligence* (Vol. 38, No. 21, pp. 23318-23325).
Zawacki-Richter, O., Marín, V. I., Bond, M., & Gouverneur, F. (2019). Systematic review of research on artificial intelligence applications in higher education–where are the educators?. *International Journal of Educational Technology in Higher Education, 16*(1), 1-27. https://doi.org/10.1186/s41239-019-0171-0