we are not just describing a Pygmalion effect where anything new (e.g. a new NetLogo model) will have a short-term positive effect but not a long-term positive effect on performance. Second, we need to conceive of abstraction in terms of not only student understanding of content in a given high school subject but abstraction as a concrete manifestation or a computational solution that can be used in subjects for problem solving. Third, students problem solving should be measured using a continuous variable and not a Boolean variable. For instance, a limitation of this study leading to measurement bias was the use of a methodology where we rated poor performance as zero and successful performance as one. CT is hardly a dichotomous variable that students either have or do not have, but a set pf complex problem solving skills. Thus CT should be tested with dynamic (cognitive and computational) models in terms of whether students are successful in using, modifying and creating code over time. Fourth, more comparative research should be conducted in terms of how well different subjects might facilitate students' development of CT. We found biotechnology, social science and chemistry equally amenable to model phenomena whereby students could learn CT. All participating teachers were able to collaborate with researchers and developers in this project to produce models that generated CT in students regardless of their subjects.

CT helps raise fundamental ontological questions in high-school teaching about what is biology, social science etc.? This was originally pointed out by Papert [1 p.140] who asked: What is the potential influence of computation on students' understanding of physics? Will CT bring student nearer to grasping what a subject is or merely confuse them about phenomena, representations, codes and models? Our teaching experiment showed that by letting students tinker with models they were able to integrate both coding, modelling and content knowledge.

ACKNOWLEDGMENTS

We are grateful to PhD Palle Nowack for engaging in fruitful discussions leading to the CMC approach. We thank professors: Michael E. Caspersen, Deborah Tatar, Keld Nielsen, Arthur Hjort, and research assistant Nick Nielsen for discussions about CT. We thank the anonymous reviewers for constructive and relevant feedback. We are grateful for financial support from Region Midt and It-vest. Last, we extend our thanks to the 15 teachers and 210 students participating in the study.

REFERENCES

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

[2] Wing, J. M. (2006). Computational thinking. *Communications of the ACM*, *49*(3), 33-35.

[3] Brennan, K., & Resnick, M. (2012, April). New frameworks for studying and assessing the development of computational thinking. In *Proceedings of the 2012 annual meeting of the American Educational Research Association, Vancouver, Canada* (Vol. 1, p. 25).

[4] Bocconi, S., Chioccariello, A., Dettori, G., Ferrari, A., Engelhardt, K., Kampylis, P., & Punie, Y. (2016, July). Exploring the field of computational thinking as a 21st century skill. In *Proceedings of the International Conference on Education and New Learning TechnologiesJuly 2016Barcelona, Spain Page* (pp. 4725-4733).

[5] Lee, I., Martin, F., Denner, J., Coulter, B., Allan, W., Erickson, J., ... & Werner, L. (2011). Computational thinking for youth in practice. *Acm Inroads*, *2*(1), 32-37.

[6] Centola, D., Wilensky, U., & McKenzie, E. (2013, April). A hands-on modeling approach to evolution: Learning about the evolution of cooperation and altruism through multi-agent modeling-The EACH Project. In *International Conference of the Learning Sciences: Facing the Challenges of Complex Real-world Settings* (p. 166). Psychology Press.

[7] Machluf, Y., & Yarden, A. (2013). Integrating bioinformatics into senior high school: design principles and implications. *Briefings in bioinformatics*, *14*(5), 648-660.

[8] Ioannidou, A., Repenning, A., Lewis, C., Cherry, G., & Rader, C. (2003). Making constructionism work in the classroom. *International Journal of Computers for Mathematical Learning*, *8*(1), 63-108.

[9] Weintrop, D., Beheshti, E., Horn, M. S., Orton, K., Trouille, L., Jona, K., & Wilensky, U. (2014, July). Interactive assessment tools for computational thinking in High School STEM classrooms. In *International Conference on Intelligent Technologies for Interactive Entertainment* (pp. 22-25). Springer, Cham.

[10] Tatar, D., Harrison, S., Stewart, M., Frisina, C., & Musaeus, P. (2017). Proto-computational thinking: The uncomfortable underpinnings. In *Emerging research, practice, and policy on computational thinking* (pp. 63-81). Springer, Cham.

[11] Lockwood, J., & Mooney, A. (2017). Computational Thinking in Education: Where does it fit? A systematic literary review. *arXiv preprint arXiv:1703.07659*.

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

[13] Caspersen, M. E., Gal-Ezer, J., Nardelli, E., Vahrenhold, J., & Westermeier, M. (2018, February). The CECE Report: Creating a Map of Informatics in European Schools. In *Proceedings of the 49th ACM Technical Symposium on Computer Science Education* (pp. 916-917). ACM.

[14] Kynigos, C. (2007). Using half-baked microworlds to challenge teacher educators' knowing. *International journal of computers for mathematical learning*, *12*(2), 87-111.

[15] Wagh, A., Cook-Whitt, K., & Wilensky, U. (2017). Bridging inquiry-based science and constructionism: Exploring the alignment between students tinkering with code of computational models and goals of inquiry. *Journal of Research in Science Teaching*, *54*(5), 615-641.

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

[17] Hjorth, A., & Wilensky, U. (2014). Re-grow Your City: A NetLogo Curriculum Unit on Regional Development. Boulder, CO: International Society of the Learning Sciences.

[18] 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.

[19] Wing, J. 2011. Research notebook: Computational thinking "What and why? ". In *TheLink Magazine, Spring. Carnegie Mellon University, Pittsburgh.*

[20] Papert, S., & Harel, I. (1991). Situating constructionism. *Constructionism*, 36(2), 1-11.

[21] Vygotsky, L. S. (1980). *Mind in society: The development of higher psychological processes.* Harvard university press.

[22] Weintrop, D., Beheshti, E., Horn, M., 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.

[23] Nowack, P., & Caspersen, M. E. (2014, November). Model-based thinking and practice: A top-down approach to computational thinking. In *Proceedings of the 14th Koli Calling International Conference on Computing Education Research* (pp. 147-151). ACM.

[24] Berland, M., Martin, T., Benton, T., Petrick Smith, C., & Davis, D. (2013). Using learning analytics to understand the learning pathways of novice programmers. *Journal of the Learning Sciences*, *22*(4), 564-599.

[25] Jang, H., Reeve, J., & Deci, E. L. (2010). Engaging students in learning activities: It is not autonomy support or structure but autonomy support and structure. *Journal of educational psychology*, *102*(3), 588.

[26] Sentance, S., & Waite, J. (2017, November). PRIMM: Exploring pedagogical approaches for teaching text-based programming in school. In *Proceedings of the 12th Workshop on Primary and Secondary Computing Education* (pp. 113-114). ACM.

[27] Sherin, B. L. (2001). A comparison of programming languages and algebraic notation as expressive languages for physics. *International Journal of Computers for Mathematical Learning*, 6, 1-61.