NetLogo Tango: Supporting Student Programming with Tangible Objects and Multi-Touch Displays

Izabel C. Olson, Michael S. Horn, Uri Wilensky
Northwestern University, Annenberg Hall, 2021 Campus Drive, Evanston, IL 60202 USA
Email: izabel@u.northwestern.edu, michael-horn@northwestern.edu, uri@northwestern.edu

Abstract: NetLogo Tango is a tool that we designed to introduce elementary school children to the NetLogo programming language. We build on the interactive properties of multi-touch tabletops and tangible objects to provide activities that we hope will be inviting and engaging and that will guide children toward progressively more powerful and expressive forms of programming. In this paper we describe our design rationale, initial prototype, and future research.

Introduction

NetLogo is a powerful agent-based modeling language that enables students and professionals alike to model complex systems and emergent phenomena (Wilensky, 1999). NetLogo is in widespread use in educational contexts from middle schools through universities, but relatively little work has been done with elementary school students. Here we describe initial work on a project called NetLogo Tango, interactive software that we created to help introduce NetLogo programming to elementary school children. Our goal is not only to encourage children to experiment with existing NetLogo models, but also to help them create their own models from scratch. To do this, we exploit interactive properties of multi-touch tabletop displays and tangible objects to help scaffold students' learning process.

NetLogo and Agent-Based Modeling

Agent-based models and simulations have become an important tool for research (Gilbert & Troitzsch, 2005), enabling scientists to explore emergent phenomena by identifying individual "agents" in a system and giving them computational rules of behavior and interaction. Undoubtedly these technologies have transformed scientific work, yet we believe that there is also potential to use agent-based modeling to enrich science, technology, and mathematics education. One major advantage of agent-based models is that they are accessible to learners without the need for great mathematical sophistication. Many important physical, biological, and social phenomena can be modeled and understood without differential equations or even advanced algebra, but rather through simple "local" computational rules such as "wolf-eats-sheep" or "sheep-eats-grass" (Wilensky & Reisman, 2006).

Design Principles

Our prior research with tangible and graphical programming languages in educational settings (Bers & Horn, 2009; Horn, Solovey, Crouser, & Jacob, 2009) has led us to adopt a hybrid design approach. By this we mean that children can create NetLogo programs using physical blocks, graphical blocks, or even text-based code. These three interaction styles are combined into a single hybrid environment in which physical, graphical, and text-based programming elements can comingle. Building on the idea of *progressive lures* (Hornecker, Marshall, & Rogers, 2007; Lidwell, Holden, & Butler, 2003) our goal is to introduce new programming concepts with an interface that is playful and inviting. After a concept has been introduced, the system will invite students to explore progressively more powerful and expressive modes of programming. Concretely, our past research in museum settings indicates that tangible interaction seems to be especially inviting to children compared to mouse-based interaction (Horn, et al., 2009). Thus, we introduce programming concepts to children with a tangible interface, one which allows the child to input commands through the handling of plastic cubes. By coupling tangible interaction with a multi-touch tabletop device, we can provide a direct transition from programming with tangible blocks to programming with graphical blocks (see figure 1).

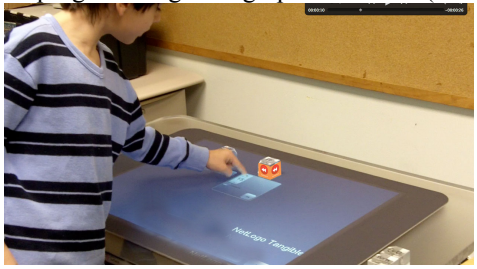

Figure 1: NetLogo Tango interface

Finally, we designed the interface to scaffold students' learning of text-based NetLogo syntax by displaying code snippets alongside both tangible and graphical commands (see figure 2). Specifically, after a child sets a parameter value for a programming block, the child can minimize the graphical element on the tabletop display. What remains is a more compact text-based representation of that command. As children look for more powerful ways to program, they can switch to text-based syntax to create and tweak programs or mix and match as they wish. A NetLogo Tango program may consist of comingled tangible, graphical, and text-based programming elements.

Figure 2: Code Snippets alongside tangible and graphical commands

Future Work

Our work with NetLogo Tango is ongoing. We will evaluate our design with a group of eight children in a summer workshop. Our goal is both to uncover usability issues and to evaluate the success of our design in terms of student engagement and learning. We expect to make substantial changes to our design both during and after the workshop. Afterward, we plan on taking NetLogo Tango into classrooms to conduct studies that will evaluate whether our design will achieve its goal of scaffolding the learning of NetLogo syntax for young children.

References

Bers, M.U. & Horn, M.S. (2009). Tangible programming in early childhood: Revisiting developmental assumptions through new technologies. In I. R. Berson & M. J. Berson (Eds.), *High-tech tots: Childhood in a digital world*. Greenwich, CT: Information Age Publishing.

Gilbert, N., & Troitzsch, K. G. (2005). *Simulation for the Social Scientist*. Berkshire, England: Open University Press, McGraw Hill Education.

Horn, M.S., Solovey, E.T., Crouser, J.R., and Jacob, R.J.K. (2009). Comparing Tangible and Graphical Programming Interfaces for use in Informal Science Education. *Proceedings of the ACM Conference on Human Factors in Computing Systems CHI'09*, (pp. 975-984). Boston, MA, USA: ACM Press.

Hornecker, E., Marshall, P., & Rogers, Y. (2007). From entry to access: how shareability comes about. *Proceedings of the 2007 Conference on Designing Pleasurable Products and Interfaces DPPI'07*, (pp. 328-342). Helsinki, Finland: ACM Press.

Lidwell, W., Holden, K., & Butler, J. (2003). *Universal Principles of Design*. Gloucester, Massachusetts: Rockport Publishers.

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

Wilensky, U. (2002). Modeling nature's emergent patterns with multi-agent languages. *Proceedings of EuroLogo,* pp. 1–6.

Wilensky, U., & Reisman, K. (2006). Thinking like a wolf, a sheep or a firefly: Learning biology through constructing and testing computational theories. *Cognition and Instruction*, 24(2), 171-209.