



Evolution in Blocks: Building Models of Evolution using Blocks

Aditi Wagh, aditiwagh@u.northwestern.edu
School of Education & Social Policy, Northwestern University

Uri Wilensky, uri@northwestern.edu
School of Education & Social Policy, Northwestern University

Abstract

Evolution undergirds the domain of biological sciences. Despite its centrality to the biological sciences, commonly used representations such as graphic simulations and cladograms, while depicting change over time, fail to provide insight into its underlying mechanisms. We have been working on developing a restructuration for evolution in the context of DeltaTick, a block-based programming interface in NetLogo. Our goal is for students to build, debug and refine models of evolutionary processes using block-based primitives in DeltaTick. We first describe the underlying motivation, and then describe the design work we have done so far. We conclude with a discussion of challenges and design tensions.

Keywords

Block-based programming, Restructuration, Modeling, Evolution

Introduction

Technology is rapidly transforming the representational basis of science education. Advances in new media technologies are making it possible for students to engage in science and mathematics with more ownership and enjoyment, and at younger ages than ever before. These advances can be leveraged to develop new representational forms or “restructurations” to encode disciplinary knowledge (Wilensky & Papert, 2006; 2010).

In this paper, we describe a restructuration for processes underlying micro- and macro-evolutionary change. This restructuration is being designed in the context of DeltaTick (Wilkerson-Jerde & Wilensky, 2010), a block-based programming interface in NetLogo (Wilensky, 1999). The motivation underlying the restructuration is twofold; one, the paucity of rich representational building environments for evolution; and two, to develop an environment in which students can engage their productive intuitive knowledge about evolution to make sense of it.

Building Models of Evolutionary Change

Evolution undergirds the ever-growing domain of the biological sciences (Gould, 2002; Kitchener, 2007). Dobzhansky's famous and often-quoted remark, “Nothing in biology makes sense except in the light of evolution” neatly captures the centrality of evolution to the biological sciences (Dobzhansky, 1973).

Popular representations of evolutionary change such as cladograms and graphic simulations have been widely studied for their impact on student learning (Ainsworth, 2009; Evans et al, 2010; Matuk & Uttal, 2009; Soderberg & Price, 2003). While these representations depict change over



time, they fail to provide insight into the mechanistic underpinnings of evolutionary change. They do not represent *how* populations change over time. However, in order for children to develop a meaningful understanding of evolution, we think it is essential they have access to the underlying mechanisms of evolutionary change to be able to make sense of this long-term phenomenon.

Agent-based models have shown great potential in helping students learn about mechanisms underlying systems by restructuring scientific disciplines such as physics (Sengupta & Wilensky, 2009), chemistry (Levy & Wilensky, 2009), material sciences (Blikstein & Wilensky, 2009) and calculus (Wilkerson-Jerde & Wilensky, in review), to make complex and inaccessible scientific content engaging, easier to visualize and learn.

Some prior work has addressed using agent-based representations to teach evolution (Centola, McKenzie & Wilensky, 2000; Wilensky & Centola, 2007). An agent-based modeling curriculum, BEAGLE (Biological Experiments in Adaptation, Genetics, Learning & Evolution), which primarily focuses on guided exploration of models, has been found to facilitate learning in eighth grade science classes (Wilensky & Novak, 2010, Wagh & Wilensky, 2012). Work done on building models of evolution has also found that it fosters deep learning (Wilensky & Centola, 2007; Xiang & Passmore, 2010).

We were interested in developing a domain-specific programming environment in which kids could quickly build models of evolution without having to do extensive programming, for several reasons. First, building fosters deep and meaningful learning (eg. Bamberger, 2001; diSessa et al, 1991; Nemirovsky & Tierney, 2001; Papert, 1980). Moreover, building models enables learners to think about the mechanisms underlying a system (Wilensky, 2003; Wilensky & Reisman, 2006). Secondly, through interviews with middle school children, we have found that children have a rich repertoire of knowledge about affordances and constraints of variations in populations that influence changes in an ecosystem (Wagh & Wilensky, 2011). Building serves as a medium for students to make their thinking visible by externalizing their mental representations (Papert, 1980; Lehrer & Schauble, 2000). The activity of building, debugging and revising models will enable learners to engage, refine and revalidate this repertoire of knowledge.

In this paper, we discuss the design of a model-building unit in evolution in DeltaTick (Wilkerson-Jerde & Wilensky, 2010), a block-based programming interface in NetLogo (Wilensky, 1999). The graphical interface of DeltaTick allows for quick constructions of models with semantic meaningfulness at the domain level. These constructions are made using domain-specific block-based primitives/procedures. Each block represents self-contained and autonomous fragments of code or procedures that function as rules of behavior for agents in the system (Kahn, 2007).

In what follows, we describe our ongoing design work. Specifically, we describe some of the new features of DeltaTick and then discuss some of the design challenges.

DeltaTick: The Design

DeltaTick was originally designed and built by Wilkerson-Jerde & Wilensky (2010). In this design project, we are doing further design work in DeltaTick to make it a felicitous environment to model evolutionary processes. The overarching goal of this project is to restructure micro and macro evolutionary mechanisms into blocks that serve as primitives for model-building. These block-based primitives represent agent-level interactions that underlie evolutionary change.

When interacting with DeltaTick, learners can build models of ecological systems by defining one or more species and/or an environment for the species. They can then assign blocks or behaviors to the species to model different kinds of agent-level interactions, run the model, and



then debug or revise it. The goal is for learners to engage in model-building to fluidly navigate between aggregate-level evolutionary change and agent-level interactions that results in that change.

Trait Blocks: Define traits & variations for a species

Species (breeds, in NetLogo parlance) created in DeltaTick can be assigned variables to model variation in a population. A variable is presented as a *Trait* and values corresponding to a variable are called *Variations*. Learners can assign *Traits* to *Species*, and define *Variations* for a *Trait*. For example, a learner can create a species, “frogs” and then define a trait such as “color of skin” and “red” and “green” as corresponding variations for the trait.

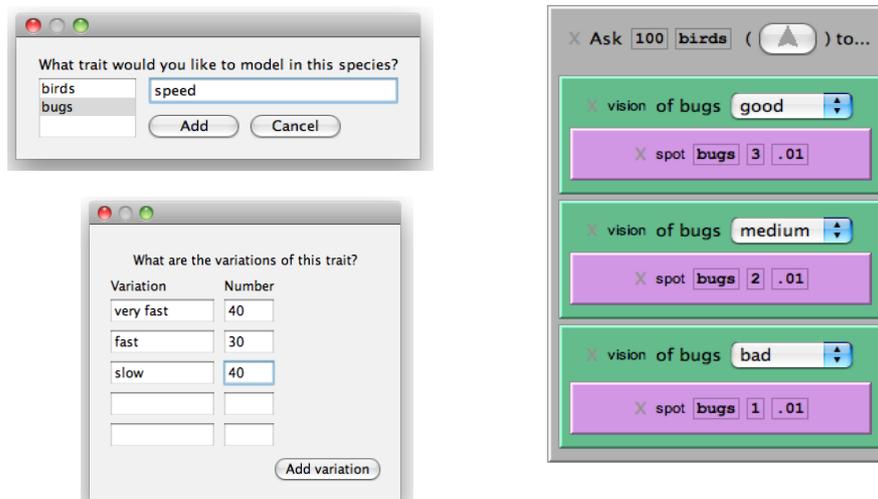


Fig 1: LEFT: Selecting a trait & variations; RIGHT: Assigning behaviors in TRAITBLOCKS

Once a trait is defined, it is represented as a TRAIT BLOCK in the environment. Each TRAIT BLOCK has a drop-down list of *Variations* defined for that *Trait*. Learners can use a TRAIT BLOCK in their model to assign behaviors with different probabilities to individuals with different variations. For instance, in the case of frogs, a learner could build a model in which Red-Frogs are eaten with a probability of .4 while Green-Frogs are eaten with a probability of .1 when the frogs live in a pond with green weed.

The goal is to enable learners to assign behaviors with different settings for individuals with different variations.

Behaviors blocks: Blocks as primitives. Blocks as procedures.

In DeltaTick, a behavior block represents a fragment of code that encapsulates a procedure. These blocks represent rules for the individuals in the world to follow. We have developed a library of blocks that, we believe, are moderately generic in their ability to capture a variety of scenarios of natural selection.

Each behavior block comes with input boxes that allow learners to specify certain values for a particular behavior. For instance, one of the blocks in the library is called CHASE. The CHASE block comes with four input options; who?, at-what-speed?, notice-within-what-distance?, and probability-of-chasing?. Similarly, SPOT is a block that is accompanied by three inputs; who?, within-what-distance?, and with-what-probability?. These input boxes enable learners to specify, at a finer-grained level, the nature of individual-level interactions. For instance, by varying the input in “within-what-distance?” of the SPOT block, learners can model variation in the ability to



sense (see, hear, smell or touch) other individuals in the environment. (see Figure 1)

We're currently working on developing libraries for other evolutionary processes such as genetic drift and speciation.

Environment Blocks

Learners can also select an environment for the species they have created from a list of pre-defined *Environments* in DeltaTick. In its current version, an *Environment* is conceptualized as a resource that provides food for the *Species* (eg. Grass). The *Environment* is represented in the form of patches in the NetLogo model. When food (grass) at any patch is consumed, the model waits for a pre-defined period of time before growing grass. *Species* and *Environments* can interact with each other. Each *Environment* comes with a library of block-based primitives specific to it (eg. Grow-grass).

Operator Blocks

When building models within DeltaTick, learners can also build an OPERATOR BLOCK. These blocks allow learners to assign behaviors to more than one group of individuals at one time. For instance, imagine a model of a fictional species, Bogsters that vary with respect to their color and strength-of-legs. A learner can define an OPERATOR BLOCK to assign behaviors to individuals who are red and short-legged. These blocks can be defined using the “and”, “or” or “neither” operator.

Challenges & Design tensions

As we continue working on the design of this environment, we find ourselves grappling with two design tensions. The first one involves designing primitives of the right size that are generative enough in their ability to build different kinds of models, and yet are meaningful to middle school children. It is very important to us that learners feel that their intuitive knowledge is relevant, and more importantly, legitimate, when building models. In alignment with this priority, our focus has been to parse mechanisms of evolutionary change into block-based primitives that will make sense to middle school children. This focus has proved to be a challenge especially in designing a library for macroevolution.

A related design tension lies in seeking a balance between making the environment open-ended so learners can build personally meaningful models, and pre-designing a library of blocks that are adequate and meaningful in the context of undetermined species and traits. These tensions continue to influence and inform our design decisions as we further develop this environment.

Acknowledgements

A majority of this work has been done in the context of learners exploring models of evolutionary change. Exploration involves running experiments with pre-built models in which the agent-level mechanisms had already been coded for the students.

References

Bamberger, J. (2001). "Turning Music Theory on its Ear: Do we hear what we see; do we see what we say." *Multidisciplinary Perspectives on Musicality: The Seashore Symposium*. Iowa City: University of Iowa Press.

Blikstein, P., & Wilensky (2009). An atom is known by the company it keeps: A constructionist learning environment for materials science using multi-agent simulation. *International Journal of Computers for*



Mathematical Learning, 14(1), 81-119.

Dobzhansky, T. (1973). Nothing in biology makes sense except in the light of evolution. *The American Biology Teacher*, 25, 125-129.

Evans, E. M., Frazier, B., Hazel, A., Kiss, A., Lane, J. D., Spiegel, A. N., & Diamond, J. (2010). Tree-thinking: Do pictorial representations of evolution help or hinder museum visitors' understanding? Understanding the Tree of Life Conference, Carnegie Museum of Natural History, Pittsburgh.

Gould, S.J. (2002). *The Structure of Evolutionary Theory*. Cambridge: Belknap of Harvard University Press.

Kahn, K. (2007). *Building computer models from small pieces*. Paper presented at the Proceedings of the 2007 Summer Computer Simulation Conference, San Diego, CA.

Kitchener, P. (2007). *Living with Darwin*. Oxford: Oxford University Press.

Levy, S. T., & Wilensky, U. (2009). Crossing levels and representations: The Connected Chemistry (CC1) curriculum. *Journal of Science Education and Technology*, 18(3), 224-242.

Matuk, C.F., & Uttal, D.H. (2009). *Interpretation, invention, and interaction: How students (mis)understand cladograms*. Paper presented at the Understanding the Tree of Life Harvard Conference. Cambridge, MA.

Sengupta, P. & Wilensky, U. (2009). Learning Electricity with NIELS: Thinking with Electrons and Thinking in Levels. *International Journal of Computing and Mathematical Learning*, 14, 21-50.

Soderberg, P., & Price, F. (2003). An examination of problem-based teaching and learning in population genetics and evolution using EVOLVE, a computer simulation. *International Journal of Science Education*, 25(1), 35-55.

Wagh, A. & Wilensky, U. (2010). Ideas-to-think-with: Useful pieces of knowledge about natural selection. In J. Clayson & I. Kalas (Eds.), *Proceedings of the Constructionism 2010 Conference*. Paris, France, Aug 10-14.

Wagh, A. & Wilensky, U. (2012). *Breeding birds to learn about artificial selection: Two birds with one stone?* In *Proceedings of ICLS 2012*, Sydney, Australia.

Wilensky, U. & Papert, S. (2010). Restructurations: Reformulations of Knowledge Disciplines through new representational forms. In J. Clayson & I. Kalas (Eds.), *Proceedings of the Constructionism 2010 Conference*. Paris, France, Aug 10-14. p. 97.

Wilkerson-Jerde, M. & Wilensky, U. (2010). Restructuring change, interpreting changes: The deltatick modeling and analysis toolkit. In *Proceedings of Constructionism 2010* (p. 97), Paris, France.

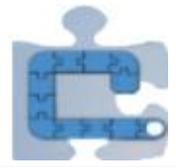
Wilensky, U. & Novak, M. (2010). Understanding evolution as an emergent process: learning with agent-based models of evolutionary dynamics. In R.S. Taylor & M. Ferrari (Eds.), *Epistemology and Science Education: Understanding the Evolution vs. Intelligent Design Controversy*. New York: Routledge.

Wilensky, U. & Novak, M. (n.d.). BEAGLE. Evanston, IL: Northwestern University, Center for Connected Learning and Computer-based Modeling.

Wilensky, U. (2003). Statistical mechanics for secondary school: The GasLab Modeling Toolkit. *International Journal of Computers for Mathematical Learning*, 8(1), 1-41 (special issue on agent based modeling).

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 (PDF). *Cognition & Instruction*, 24(2), pp. 171-209.

Wilensky, U. & Resnick, M. (1999). Thinking in Levels: A Dynamic Systems Perspective to Making



Sense of the World. *Journal of Science Education and Technology*, 8(1).

Xiang, L. & Passmore, C. (2010). *The Use of Agent-based Programmable Modeling Tool in 8th Grade Students' Model-Based Inquiry*. *The Journal of the Research Center for Educational technology*, 6(2), 130-147.