Chapter 2
MaterialSim: A Constructionist Agent-Based Modeling Approach to Engineering Education

Paulo Blikstein and Uri Wilensky

Introduction

For the past two decades, the engineering education community has started to come to terms with an unfortunate paradox: despite a view of engineering as the ultimate design profession, very little actual experience in design is incorporated into undergraduate engineering curricula. Recently, pressured by decreasing enrollment, unmotivated students, and an avalanche of new demands from the job market, several engineering schools have started to roll out ambitious reform programs, trying to infuse engineering design into the undergraduate curriculum. A common element in those programs is to introduce courses in which students design products and solutions for real-world problems, engaging in actual engineering projects. These initiatives have met with some success and are proliferating into many engineering schools. Despite their success, they have not addressed one key issue in transforming engineering education: extending the pedagogical and motivational advantages of design-based courses to theory-based engineering courses, which constitute the majority of the coursework in a typical engineering degree, and in which traditional pedagogical approaches are still predominant.

In this chapter, we describe and analyze a series of studies designed to address this exact issue, in which we investigate undergraduate students' learning of

P. Blikstein (✉)
Learning Sciences and Technology Design Program, School of Education and (by courtesy) Department of Computer Science, School of Engineering, Stanford University, Stanford, CA, USA
e-mail: paulob@stanford.edu

U. Wilensky
Center for Connected Learning and Computer-Based Modeling, Department of Learning Sciences, School of Education and Social Policy and Department of Computer Science, McCormick School of Engineering, Northwestern Institute on Complex Systems, Northwestern University, Evanston, IL, USA

This chapter is based on Blikstein and Wilensky (2009).

M.J. Jacobson and P. Reimann (eds.), *Designs for Learning Environments of the Future: International Perspectives from the Learning Sciences*, DOI 10.1007/978-0-387-88279-6_2,

theoretical content in materials science through designing (i.e., programming) their own computer models of scientific phenomena. Our research design emerged from extensive classroom observations followed by a literature review of engineering and materials science education, as well as analysis of class materials, and interviews with students. Our observations (consistent with the literature review) indicated that students' understanding of the subject matter was problematic, and that the teaching was not up to the challenge of the sophistication of the content. Based on this diagnosis, we have iteratively designed constructionist (Papert, 1980) model-based activities for materials science - *MaterialSim* (Blikstein & Wilensky, 2004a; 2004b, 2005a; 2005b; 2006a; 2008) - a suite of computer models, learning activities, and supporting materials designed within the approach of the complexity sciences and agent-based modeling. The activities were built within the NetLogo (Wilensky, 1999b) modeling platform, enabling students to build models, and investigate common college-level topics such as crystallization, solidification, crystal growth, and annealing.

The studies consist of both design research and empirical evaluation. Over 3 years, we conducted an empirical investigation of an undergraduate engineering course using MaterialSim, in which we investigated: (a) The *learning outcomes* of students engaging in scientific inquiry through interacting with MaterialSim; (b) The *effects of students programming their own models* as opposed to only interacting with preprogrammed ones; (c) The *characteristics, advantages, and trajectories of scientific content knowledge* that is articulated in epistemic forms and representational infrastructures unique to complexity sciences; and (d) The *design principles* for MaterialSim: what principles govern the design of agent-based learning environments in general and for materials science in particular? Twenty-one undergraduates enrolled in a sophomore-level materials science course participated in three studies in 2004, 2005, and 2006, each comprised of a survey, preinterview, interaction with the prebuilt computer models, students' construction of new models, and a postinterview.

2.5 Min per Equation

Our classroom observations suggested that the ever-growing sophistication and extent of college-level content in engineering (and, in particular, materials science) pose a difficult challenge to current teaching approaches. One reason is that the important equations and mathematical models taught in undergraduate materials science courses are not only complex, but are connected in nontrivial ways to multiple sets of other theories, concepts, and equations. Teachers end up resorting to multiple equations and models to derive and explain a single canonical phenomenon, and those equations and formulas are oftentimes located in a different areas of mathematical modeling (statistical mechanics and geometrical modeling, for example). What is more, many "engineering theories" are combinations of empirical models or approximations, and not pristine, rigorous, and easy-to-describe theories.

As a result, what takes place in a typical engineering theory course lecture is not a linear progression of equations, from simple to complex. Conversely, when a new phenomenon is taught to students, a very large number of new connections with previously learned topics will likely arise on multiple levels, generating even more specialized equations to account for those connections. The sheer number of equations generated makes a comprehensive exploration infeasible in the classroom. Our classroom observations revealed that, in a typical 30-minute period, students would be exposed to as many as 12 unique equations with 65 variables in total (not counting intermediate steps in a derivation) – or approximately 2.5 minutes for each equation and 45 seconds for each variable!

This overloading with equations and variables seems a likely candidate for explaining the students' difficulties described above. We decided to investigate this hypothesis and investigate: what kind of understanding did this multiplicity of explanation levels and the "overloading" of equations foster in students? In addition to understanding the consequences of the traditional pedagogical approaches, we wanted to explore possibilities of an alternate approach, and examine the consequences of using agent-based models (Collier & Sallach, 2001; Wilensky, 1999a; Wilensky & Resnick, 1999) enacted as microworlds (Edwards, 1995; Papert, 1980) for students' understanding of materials science content since our previous research suggested that using such a modeling approach might be a better match of content to student cognition.

The agent-based modeling approach, as we will explain in detail, enables modelers to employ simple individual-level rules to generate complex collective behaviors. These simple rules capture fundamental causality structures underlying complex behaviors within a domain. Wilensky, Resnick, and colleagues (Wilensky, 1999a; Wilensky & Reisman, 2006; Wilensky & Resnick, 1999) have pointed out that such rules could be more accessible to students than many of the equations describing the overall, macroscopic behaviors of a system. The agent-based approach is also a better fit with the constructionist pedagogical framework (Papert, 1991). The history of constructionist pedagogy has included three principal modes of learner activity: (a) designing and programming computational artifacts (programming-based constructionist activities – PBC); (b) exploring computer-based microworlds (microworlds-based constructionist activities – MBC); and (c) engaging in the first two modes with computationally augmented physical structures (tangible-based constructionist activities – TBC). Agent-based modeling can be used in any of these three modes. In the second mode, models can function as constructionist microworlds, as agent-based models can represent the underlying logic of a system, enabling students to investigate and modify features of that structure and explore the consequences of those changes, and through that exploration and investigation come to understand the domain. In the first mode, students design and program their own agent-based models and gain a deep sense of the design space of domain models. In the third mode, students can connect physical sensors and motors to agent-based models and let the models take input from real world data and drive real world action (*bifocal modeling*, Blikstein & Wilensky, 2007). In the MaterialSim project, we have designed artifacts and activities to engage

students in each of these three modes. In this chapter, we will explore the first two modes, i.e., microworlds-based (MBC) and programming-based constructionist activities (PBC).

The conjecture that using agent-based modeling (ABM) would be a better cognitive match for students is based on research that suggests that this approach fosters more generative and extensible understanding of the relevant scientific phenomena. In the case of materials science, instead of multiple models or numerous equations, this framework focuses on a small number of elementary behaviors that can be applied to a variety of phenomena. Instead of a *many-to-one* approach (many equations to explain one phenomenon), we attempt here a *one-to-many* approach (one set of local rules to explain many phenomena), through which students would see diverse materials science phenomena not as disconnected one from the other, but rather as closely related emergent properties of the same set of simple atomic or molecular rules. A second major focus of our study was to determine: What kind of understanding do students develop of the materials science content when they study it from this agent-based, *one-to-many* perspective?

In addition to those two driving questions, we wish to explore one further dimension of this pedagogical approach. There have been several recent studies of students using ABM to learn science; in many of these studies the approach taken was to design sequences of models and microworlds for students to explore (e.g., Levy, Kim, & Wilensky, 2004; Stieff & Wilensky, 2003). We extend this approach to the domain of materials science but mainly we wish to find out what the effect will be from *moving beyond microworlds and enabling students to choose phenomena of interest to them and construct their own models* in the domain of material science (for another such approach, see Wilensky & Reisman, 2006).

In this chapter we are focusing on the interviews and laboratory studies prior to the classroom implementation (subsequent design experiments on classroom implementations are reported in Blikstein, 2009). We report on a particular pedagogical design and present evidence in the form of excerpts and samples of students' work, which demonstrates that the experience with MaterialSim enabled students to identify and more deeply understand unifying scientific principles in materials science, and use those principles to effectively construct new models.

Materials science is one of the oldest forms of engineering, having its origins in ceramics and metallurgy. In the nineteenth century, the field made a major advance when Gibbs found that the physical properties of a material are related to its thermodynamic properties. In the early twentieth century, the field of materials science concentrated on metals and university departments were often called "metallurgical engineering departments." The field has since broadened to include polymers, magnetic materials, semiconductors, and biological materials and since the 1960s has been called materials science. Today, with the explosion of research in nanotechnology, alternative energy, and new materials, it has gained a very significant role in the realm of technological innovation. However, the teaching of materials science has not kept up with the rapid advances in the field. Therefore, before diving in to the study, we step back and contextualize the teaching of materials science within the landscape of engineering education, its recent critique, and calls for reform.

A New Scenario in Engineering Education

In 2007, approximately 400,000 students took college-level engineering courses in the United States alone (American Society for Engineering Education, 2007). As early as the 1960s, education researchers (Brown, 1961; Committee on the Education and Utilization of the Engineering, 1985; Jerath, 1983; MIT Center for Policy Alternatives, 1975; Panel on Undergraduate Engineering Education, 1986) have pointed out that engineering education lags behind in its adoption of newer approaches to teaching and learning. In recent years, there have been numerous calls for reform from the engineering education community and several schools have implemented reform initiatives (Einstein, 2002; Haghighi, 2005; Russell & Stouffer, 2005). The driving force behind engineering education reform programs were both new societal needs (Dym, Agogino, Eris, Frey, & Leifer, 2005; Committee on the Education and Utilization of the Engineering, 1985; Katehi et al., 2004; Tryggvason & Apelian, 2006) and technical advances. As basic science and engineering become increasingly intertwined in fields such as nanotechnology, molecular electronics, and microbiological synthesis (Roco, 2002), students and professionals have to deal with time scales from the nanosecond to hundreds of years, and sizes from the atomic scale to thousands of kilometers (Kulov & Slin'ko, 2004). This wide range of subjects and problems makes it prudent not to try to cover all the relevant knowledge so that students master the knowledge in each domain, but instead to help students develop adaptive expertise (Bransford & Schwartz, 1999; Hatano & Oura, 2003) that they can apply to new problems and situations.

However, most engineering curricula remain in coverage mode – curricula are still so overloaded with transient or excessively detailed knowledge that there is no time for fostering students' fundamental understanding of content matter (Hurst, 1995). This phenomenon of curricular overloading is not exclusive to higher education. Tyack and Cuban (1995) identified the "course adding" phenomenon in most of twentieth century reform initiatives across all levels of education – new courses are regularly added to the curriculum to satisfy new societal needs. However, the situation becomes more problematic as we envision engineering schools in two or three decades from now. At some point the limit is reached and if courses need to be added, others must be removed – but can we afford to exclude anything from the curriculum? A major challenge is in how to go about deciding what courses can be dispensed with (and what knowledge).

A common approach in many universities has been to add hands-on engineering design courses to the curriculum. Design-based courses represented one attempted solution to the overcrowding of courses as they enable multiple content domains to be taught together. Design courses have been highly successful (Colgate, McKenna, & Ankenman, 2004; Dym, 1999; Dym et al., 2005; Lamley, 1996; Martin, 1996; Newstetter & McCracken, 2000), but they are not the universal answer for all problems afflicting engineering education. First, a significant part of engineering education consists of basic science (physics, chemistry), engineering

science (fluid mechanics, thermodynamics), and mathematics (calculus, linear algebra). It is challenging for design-based courses to focus on the core conceptual elements of these highly theoretical knowledge domains as the physicality of students' projects can be an obstacle for learning invisible or microscopic phenomena such as chemical reactions, pure mathematics, or quantum physics. Secondly, the technological tools used in those reform initiatives (such as modeling and design software) are the same employed by professional engineers in their everyday practice and not especially designed for learning. Using professional-based tools might be tempting as they enable students to achieve more rapidly the desired engineering design. In the specific case of materials science, however, this might not be the best choice. Most software tools used in engineering courses do not afford insight into the computation underlying their design and functioning. For engineering practice, indeed, a tool has to yield reliable and fast results – understanding what's "under the hood" is not necessarily useful. However, in materials science, this could be disadvantageous for learners. The computational procedures might embody an essential, perhaps crucial, aspect of the subject matter – *how* the conventional formulas and representations capture the phenomena they purport to model. Manifestly, no computer-modeling environment can uncover all of its computational procedures – it would be impractical example, to have students wire thousands of transistors to understand the underlying logic of the modeling environment. Nevertheless, we believe that most of these environments could be made profitably more transparent to students. However, the epistemological issues regarding the tools and knowledge representations in traditional engineering teaching run deeper.

First, in materials science, many of the traditional formulas *themselves* are opaque – they embody so many layers of accumulated scientific knowledge into such a complex and concise set of symbols that they do not afford common-sense insight and grounding of the causal mechanisms underlying the phenomena they purport to capture. Different from the basic sciences, engineering knowledge is a complex matrix of empirical "engineering laws," theories derived from fundamental mathematical or physical models, approximations, and rules of thumb. Making sense of this complex matrix is challenging for novices. Although using formulas and conventional engineering representations is perhaps conducive to successful *doing* (designing a new alloy, for example) it does not necessarily lead to principled understanding (knowing how each of the chemical elements interact and alter the properties of the alloy.[1]) Particularly, we are interested in "*extensible" understanding* – learning principles from one phenomenon that could be applied to other related phenomena.

Secondly, there is an important distinction to be made in how representations relate to the phenomena they purport to describe. We are not arguing that aggregate equational representations are intrinsically ill suited for learning engineering or science as there are many cases in which equational representations are fruitful for

[1] For more on design for learning versus design for use see, for example, Soloway, Guzdial, & Hay, 1994.

learning. Sherin (2001), for example, showed how the symbolic manipulation of formulas could lead to a gain in conceptual understanding in physics.

We are arguing that in some cases aggregate equations can hide important information needed for learning. In some areas of science, equations are directly postulated at the macro level, i.e., they are not necessarily an aggregation of simpler, local behaviors, or the microscopic behaviors are not relevant to the phenomenon under scrutiny. For example, in simple Newtonian motion, we are interested in predicting the motion of bodies, but looking at the individual atoms of the body might not offer additional insight into the phenomenon – the macroscopic and microscopic behaviors could be analogous, i.e., the body and its atoms would be moving in the same fashion. In such areas, aggregate equations reveal most of the needed information. In other domains, however, the opposite is true: equations are an aggregation of microscopic behaviors, and those offer fundamental insights into the phenomenon, and are not analogous to the aggregate equations (for example, statistical mechanics, or diffusion). Therefore, for the latter categories of phenomena, aggregate equational representations might generate an *epistemological gap* (Blikstein, 2009) – the mathematical machinery needed to derive macro behaviors from microbehaviors is intricate, and rigorous mathematical frameworks to guide such work are still being developed (see, for example, Parunak, Savit, & Riolo, 1998; Yamins, 2005; Wilkerson-Jerde & Wilensky, 2009). This *epistemological gap* makes it difficult to keep track of how micro- and macro-level parameters are related and influence each other, or to understand how intuitive, simple microbehaviors are represented in aggregate analytical forms. Our research, indeed, suggests that an exclusive use of equational representations for those types of phenomena can constitute an obstacle for students in acquiring conceptual understanding in domains of engineering in which the interaction of microscopic entities is at the core of the content matter. For those phenomena, in which equational representations show an aggregation of microbehaviors, it seems to be especially beneficial to unpack and deconstruct the traditional aggregate representations, restructuring domains of knowledge around the study of local, individual, "nonaggregated" phenomena (Wilensky et al., 2005; Wilensky & Papert, in preparation; diSessa, 2000).

For the most part, however, professional engineering tools whose main goal is arriving at results rather than uncovering processes emphasize aggregate-level simulation to predict macroscopic variables (Wilensky, 1999a; 2003). However, the focus on microbehaviors could make such content intrinsically more learnable and accessible. For example, *temperature* is a macroscopic, aggregate description of a microscopic state of individual molecules (their speed or energy), just as *pressure* is an aggregation of the number of collisions between gas molecules and the walls of the container. At an *aggregate* level, those variables are dependent on a number of different events and phenomena, and thus numerous equations and models have to be employed to predict them, oftentimes "mixing-and-matching" different levels of explanation and mathematical modeling approaches. On the other hand, at the microscopic level, the number of events and phenomena influencing a local interaction is dramatically lower than at an aggregate level, because many of the variables observed macroscopically are emergent properties of the local behaviors.

In this chapter, we describe a learning design framework that benefits from this fact, focusing on simple agent-level behaviors (i.e., atomic- and molecular-level interactions) from which complex macroscopic behaviors emerge. We believe that this framework is especially useful in a scenario of increasing technological complexity and specialization. Materials science has transformed itself considerably over the last decade, with the advent of nano- and biomaterials, as well as the explosion of computational materials science as a core research strand. The number of materials, alloying elements, fabrications techniques, and industrial applications has grown so quickly and vastly that "covering" all the knowledge by simply adding new information to the curriculum would be infeasible. Additionally, the high level of abstraction that the new advances in materials science are bringing makes it increasingly difficult to give students any real world "feel" for the ideas learned in the classroom, as well as clear connections with their previous knowledge. While many archetypal problems in introductory physics would involve one falling body or two colliding objects, typical undergraduate problems in materials science involve simultaneous interactions of billions of atoms. Those interactions generate cascading effects that are hard to predict or understand with conventional mathematical equations, or any real-world intuitions. We posit that the microbehaviors are easier to understand and model, and could be connected to previous knowledge and intuitions about how individual people or physical bodies behave (Wilensky, 1999a). Thus, unifying, behaviors embedded in agent-based models are helpful for acquiring solid understanding of these principles, which bridge the micro- and macrolevels (Wilensky & Resnick, 1999). Consequently, we argue that the new computational tools should not be simple add-ons to the present curriculum, but part of their backbone – eventually restructuring the encoding of the content matter itself. In this, we follow the framework of Wilensky and Papert and their coinage of the word "restructuration," (Wilensky et al., 2005; Wilensky, 2006; Wilensky & Papert, in preparation) to refer to the reencoding of knowledge in an alternate representational system.

Our approach is one attempt in this direction. It builds up from previous research on the use of multiagent simulation tools in schools to investigate a wide range of phenomena. Wilensky and Resnick (1999) first noted the need to pay attention to "levels" and possible "slippages" between them, and highlighted the importance of the understanding of emergent behaviors for learning science. Wilensky, Papert, and colleagues have argued that computational representations have reached a point of development where we can embark on a program of radical "restructuration" of the science curriculum using these representations (Wilensky et al., 2005; Wilensky & Papert, in preparation). Goldstone and Wilensky (2008) have called for such a restructuration of science curricula using common transdisciplinary "patterns" such as energy minimization, positive feedback, and simulated annealing. In terms of implementation in school and universities, over the past decade and a half, educators have successfully employed agent-based modeling in undergraduate chemistry (Stieff & Wilensky, 2003), high-school chemistry (Levy et al., 2004; Levy, Novak, & Wilensky, 2006), probability, and statistics (Abrahamson & Wilensky, 2005; Wilensky, 1995), robotics

(Berland & Wilensky, 2004, 2005), physics (Sengupta & Wilensky, 2008; Wilensky, 1993; 1999a; 2003), evolution, population dynamics, and mathematics (Centola, Wilensky, & McKenzie, 2000; Wilensky, Hazzard, & Longenecker, 2000; Wolfram, 2002). Ironically, despite the widespread use of agent-based modeling in materials science, we have not found significant research investigating the use of such models for learning and teaching materials science.

We will present and discuss a series of three laboratory studies of a computer-based learning environment which addresses the aforementioned challenges by offering students opportunities to build their knowledge by designing computer models based on simple computational behaviors. The user studies were comprised of classroom observations, pre/post interviews, pre/post surveys, and data analysis from individual sessions with students using the designed materials.

Before diving into the study, some background information on materials science content and teaching is necessary to illustrate the differences between traditional and the agent-based representations. As the divergences in representation are at the core of this study, the next section will be dedicated to describing these two representations and how they differ. This will prepare the way for the description of our design and data analysis.

Equational vs. Agent-Based Methods in Materials Science

Grain Growth: A Central Phenomenon in Materials Science

Most materials are composed of microscopic "crystals." Even though we commonly associate the term "crystal" with the material used in glassware manufacturing, its scientific use is different. A crystal is an orderly arrangement of atoms, a regular tridimensional grid in which each site is occupied by an atom. In materials science, scientists use the term "grain" to refer to such an arrangement. Most materials are composed of millions of these microscopic grains, and their average size is one of the most important characteristics of a material, contributing to, among other properties, strength, toughness, and corrosion resistance. For example, a car built with steel with a wrong grain size could just break apart during normal use, or be destroyed even in a minor accident. However, grain size can change, too – high temperature is the main driving force. This phenomenon, known as *grain growth*, is exhaustively studied in materials science: small grains disappear while bigger ones grow (the overall volume is maintained). Airplanes turbines, for instance, can reach very high temperatures in flight – an incorrectly designed material could undergo grain growth and simply break apart. The photographs in Fig. 2.1 (magnified 850×) show typical results after 20 h under 900ºC.

Because grain growth is such a central phenomenon in materials science, and since it is an excellent example of how the same phenomena can have two different – and correct – representations, in what follows we will describe in detail these two representations.

Equational Representation of Grain Growth

In this section we describe the classical approach to modeling grain growth in materials science. This approach primarily employs mathematical equations. For those who want to skip over the mathematical details, it is sufficient to note that the classical approach makes use of several non-trivial equations to describe the phenomenon.

Burke (1949) was one of the first to introduce a law to calculate grain growth and proposed that the growth rate would be inversely proportional to the average curvature radius of the grains:

$$R = kt^n$$

where R is the mean grain size of the grains at a given time, t is time, k is a constant that varies with temperature, and n depends on the purity and composition of the material, as well as other initial conditions.[2]

In other words, Burke's law states that large grains (lower curvature radius) grow faster, while small grains (high curvature) have slower growth, or shrink. The mathematical formulation of Burke's law also reveals that, as grains grow, the growth rate decreases. A system composed of numerous small grains (see Fig. 2.1, left) would have a very fast growth rate, while a system with just a few grains (see Fig. 2.1, right) would change very slowly. One of Burke's approximations was to consider grains as spheres with just one parameter to describe their size (the radius). For most practical engineering purposes, this approximation yields acceptable results – however, as we previously discussed, its practical efficacy does not necessarily mean that this approach is the best way to understand the phenomenon. Due to the applied and integrative aspect of engineering research and practice, oftentimes explanations are drawn from a variety of sources: empirical equations, geometrical proof, thermodynamics, algebraic deductions, or statistical mechanics.

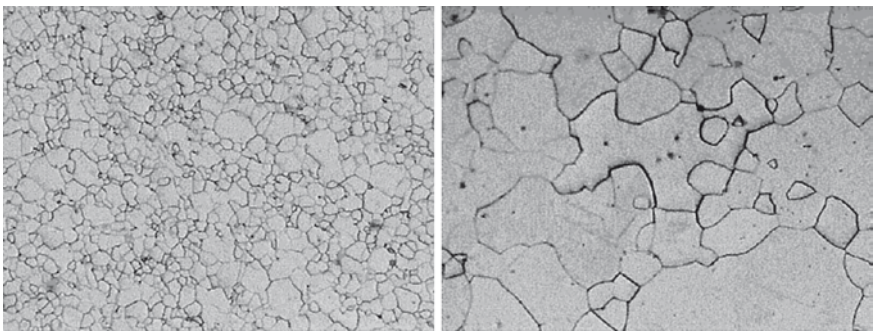


Fig. 2.1 Metallic sample before and after grain growth (Blikstein & Tschiptschin, 1999)

[2] Its theoretical value is 0.5 for pure materials under ideal conditions.

Our classroom observations revealed that, for example, when explaining grain growth and deriving Burke's law, at least three sources were employed during the classes covering the phenomenon:

- The *Laplace–Young equation for pressure*, which is commonly used in fluid dynamics to calculate surface tension in liquid–gas interfaces.
- *The flux equation*, based on statistical mechanics, which calculates the probability of atoms to move around the material.
- Geometrical approximations, which makes it possible to assume that grains or impurities in the materials are perfect spheres.

A detailed account of these equations is given elsewhere (Blikstein, 2009). We refer to this pedagogical approach as "many-to-one": many models and equations to describe one phenomenon. Our research suggests that although the many-to-one modeling approach in useful in the hands of experienced engineers in real-world situations, or very skilled researchers with high mathematical skills, this multitude of models can be an obstacle to student understanding. The mathematical machinery needed to weave together the geometrical approximations, the Laplace–Young equation, and the flux equation is very elaborate, and to achieve the simplicity and elegance of Burke's law, many assumptions and simplifications were made by the instructor. What is more, the resulting derivations and equations are specific to canonical cases, and the introduction of additional variables (for example, impurities, or temperature gradients) requires an even greater mathematical sophistication.

Agent-Based Representation of Grain Growth

Apart from equational models, heuristics (engineering "rules of thumb") are also important instruments for engineering practice. For example, when explaining grain growth, teachers commonly resort to a classic rule of thumb: large grains grow and small grains shrink. However, despite the usefulness of such heuristics to help students gain intuition into particular topics, they are not very generalizable, do not have a formal representation, and are usually domain-specific. The "large grains grow, small grains shrink" rule of thumb, for example, was shown to be particularly inaccurate when, in the early eighties, scientists started to use computer simulation as a research tools in materials science. Anderson, Srolovitz, Grest, & Sahni (1984) proposed the widely known theory for computer modeling of grain growth using a multiagent-based approach (then referred to as the "Monte Carlo method"). This kind of simulation not only made predictions faster and more accurate, but also allowed for a completely new range of applications. Researchers were no longer constrained by approximations or general equations, but could make use of actual atomic behaviors and realistic geometries. As stated by Srolovitz, Anderson, Sahni, and Grest (1984):

> While it is generally observed that large grains grow and small grains shrink, instances where the opposite is true can be found. [...] The results indicate the validity of a random walk description of grain growth kinetics for large grains, and curvature driven kinetics for small grains. (p. 796)

In other words, Srolovitz *et al.* state that the classic rule of thumb for grain growth ("large grains grow, small grains shrink") is not always valid, and that randomness plays an important role. Given the microscopic dimensions and small time scale of the phenomenon, practically the only way to visualize this new finding is through computer simulation. In contrast, the *traditional* methods for investigating grain size and growth reflect the tools (and visualization techniques) that were available in the 1950s: mathematical abstractions, geometrical modeling, approximations, and empirical data. These traditional methods and techniques, having become the methods of choice to explain the phenomena, made their way to textbooks and classrooms, and thus were established as the mainstream path to study grain growth.

Agent-based simulation of grain growth offers a different perspective. Its principle is the thermodynamics of atomic interactions, which is a simple and powerful model with explanatory power covering a wide range of phenomena. The first step is to represent the material as a 2D matrix, in which each site corresponds to an atom and contains a numerical value representing its crystallographic orientation (the angle of orientation of the atomic planes in one particular grain compared to an arbitrary fixed plane). Contiguous regions (with the same orientation) represent the grains. The grain boundaries are fictitious surfaces that separate volumes with different orientations. The stability of each atom is the matrix depends on the number of different neighbors around it. The more different neighbors one atom has, the more unstable it is, and more likely to migrate to a different location. The algorithm is comprised of the following steps:

- Each atom of the matrix has its energy[3] calculated based on its present crystallographic orientation (*2*) and the crystallographic orientation of its neighborhood – the more neighbors of differing orientation, the higher the atom's energy. Figure 2.2 (left side) shows the central atom with four different neighbors, hence the value of its *initial energy* is **4**.

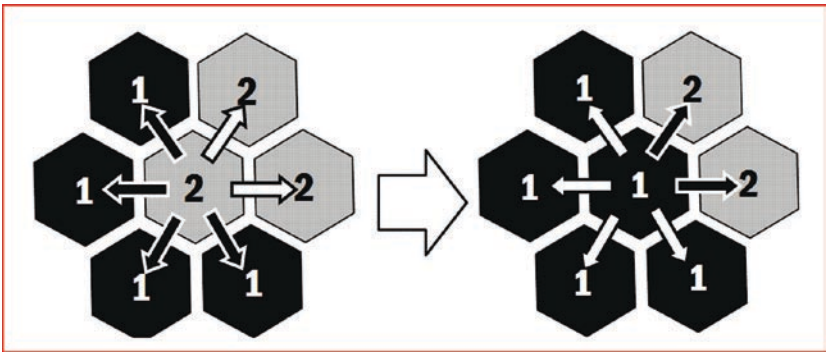


Fig. 2.2 Initial and final free-energy calculations. *Black* and *white arrows* denote different or equal neighbors

[3] Although the technical term would be "free energy," for simplicity we will use "energy."

- One new random crystallographic orientation is chosen for that atom among the orientations of its neighbors. In this case, as observable in Fig. 2.2, the current value of the central atom is **2**, and the attempted new value is **1**.
- The atom's energy is calculated again, with the new proposed crystallographic orientation (**1**). Fig. 2.2 (right side) shows that there are only two different neighbors in the new scenario, thus the *final energy* decreases to **2**.
- The two states are compared. The value that minimizes the energy is chosen. In this case, the initial energy was **4** and the new energy is **2**, so the latter value is lower and constitutes a state of greater stability. Therefore, the more different neighbors one has, the less stable one is, and thus more inclined to switching to a different orientation.

The agent-based approach captures the intricacy of the phenomenon with a single parsimonious model. In addition to the elegant simplicity of this model, it embodies the one-to-many modeling framework as it may also be used generatively to understand other phenomena as well, such as diffusion or recrystallization. The agent-based model of grain growth has been extensively tested and verified, and shown to achieve the same results as its aggregate, equational counterpart (Anderson et al., 1984; Srolovitz et al., 1984).

Unfortunately, even though new computational research tools are enabling researchers in materials science to accelerate scientific discovery and explore uncharted territory, computational methods have not yet reached mainstream engineering classrooms. Thornton and Asta (2005) conducted a comprehensive survey about the state of computational materials science in undergraduate and graduate courses at the 20 leading programs in the United States. Whereas many universities are creating or planning to initiate computational materials science courses, the prevailing mindset is that students should learn modeling *after* learning the "science." In other words, computer modeling is regarded as "icing in the cake" to take place after the "real" scientific understanding is achieved. Our work, in contrast, evaluates the usefulness of a different approach: learning the science *by* modeling.

Grain growth is a prototypical example. In the previous sections, we described how it is common practice to teach students to consider grains as spheres (which they are not), grain boundaries as real entities (whereas they are just imaginary lines between grains), and to make use of numerous metaphors and rules of thumb (e.g., "big grains swallow small grains," "particles hold boundaries," etc.) to describe and predict changes in the material.

Both traditional methods and computer-based methods of investigating grain growth rely on *modeling*. The scientific enterprise is the process of creating useful approximations to help us understand critical or interesting properties of reality (see, for example, Pagels, 1988). Models from each historical period reflect the tools available at that time. The example of grain growth is illustrative of a common practice in many fields of academic research, in particular engineering. The availability of certain technologies for research shapes how researchers approach a certain problem, and the subsequent "encoding" of the knowledge is heavily influenced by those technologies. As the initially empirical or exploratory hypothesis gradually transitions to becoming a full-blown theory, they transmit much of those

influences to the theories themselves, and consequently to the curricula. In this chapter, we argue that the current "encoding" of the knowledge about grain growth, and materials science in general, is a function of the available research technology, and the state of the field, and not an intrinsically superior way of encoding knowledge. In the following section, we describe the software infrastructure used in the project, and the design of the MaterialSim models.

Software Design: NetLogo and MaterialSim

NetLogo

NetLogo (Wilensky, 1999b) is a direct descendant of the Logo language (Papert, 1980). It is a freely available, integrated multiagent modeling environment, designed and developed by the second author at Northwestern University's Center for Connected Learning and Computer-Based Modeling. It includes a graphical user interface for exploring, experimenting with, and visualizing models, as well as a multiagent modeling language (MAML) used for authoring models (see Fig. 2.3). Such languages enable users to easily create and manipulate numerous computational entities ("agents") and define simple rules that govern their behavior. For example, to create 100 agents (or "turtles," in NetLogo's lingo) on the computer screen, the user has to simply type:

```
create-turtles 100
```

To make all of those 100 turtles move 10 units forward, users would type:

```
ask turtles [forward 10]
```

Users can also define simple rules that govern the behavior of the agents. NetLogo agents can perform simple rule-based behaviors, such as to seek being surrounded by agents with similar properties, or to avoid areas already occupied by other agents. For example, to ask all turtles to check for neighbors (within a one-patch[4] radius) and move backwards 10 units in case there are at least four neighbors around, we would use the following command:

```
ask turtles [if (count neighbors in-radius 1) > 4 [back 10]]
```

Such simple agent rules, however, may give rise to complex emergent *aggregate* phenomena, many of which are congruent with their traditional macroscopic formula-based descriptions. In addition to the modeling language itself, NetLogo includes a graphical user interface with advanced visualization features, such as multiple topologies and 3D representations. It also includes some specialized tools such as BehaviorSpace (Wilensky & Shargel, 2002), which enables users to explore

[4] The NetLogo world (or screen) is divided into a grid of square cells called patches. The size of the patches can be defined by the user.

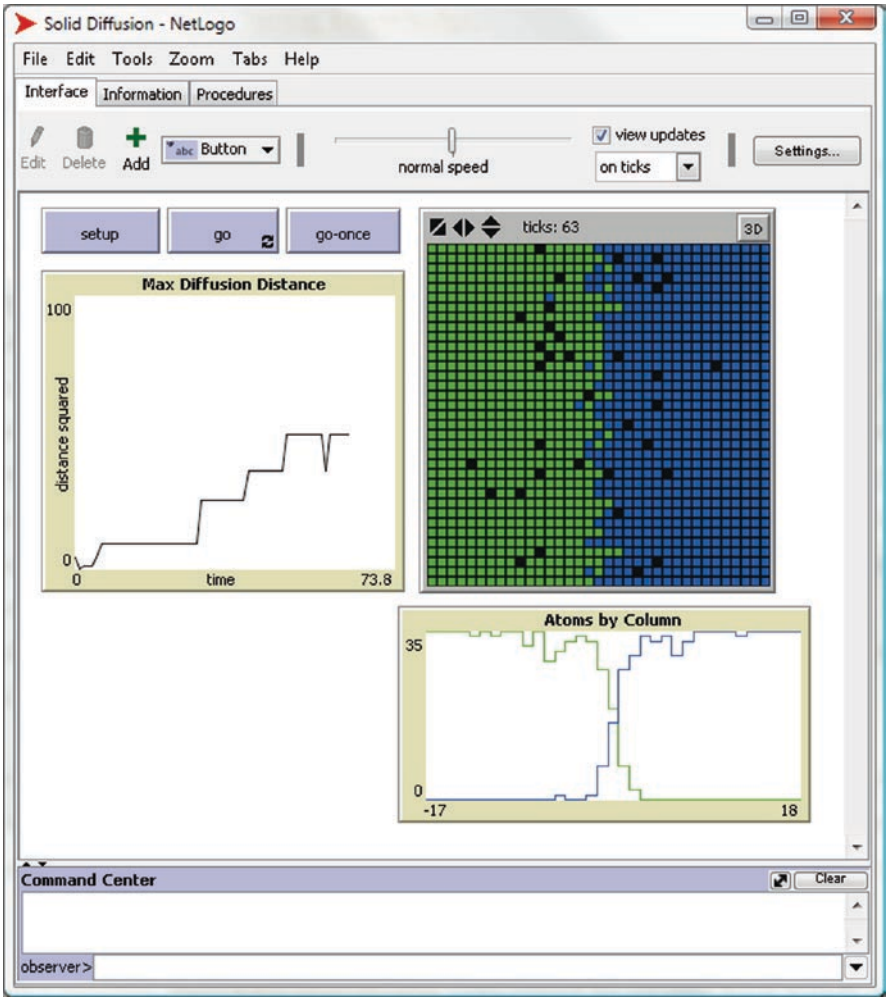


Fig. 2.3 The NetLogo modeling environment, with a "Solid Diffusion" model

a wide parameter space by running multiple experiments, and automatically logging the data. NetLogo comes with a large library of sample models as well as model-based curricula.

MaterialSim

We chose the NetLogo modeling-and-simulation environment as a platform as it is well adapted to the activities of the studies, in particular, NetLogo's "low-threshold, no-ceiling" (Papert, 1980; Tisue & Wilensky, 2004; Wilensky & Rand,

2009) design enables learners to achieve sophisticated results within a relatively short period of time, and its built-in visualization tools allow dynamic, flexible, and customizable views. MaterialSim is a set of models and activities built by the authors within the NetLogo environment, for investigating materials science phenomena such as crystallization, solidification, casting, grain growth, and annealing. MaterialSim's design is different from many other curriculum projects where instructional designers often prepare a series of models for students. MaterialSim, instead, focuses on students *programming* models. This design choice was based on previous research and on the learning goals of the project. For example, previous studies on students programming their own agent-based models report that participants were able to infer behaviors based on incomplete or sparse information, as well as gain deep understanding of how changes in microbehaviors can alter a given system (e.g. Berland & Wilensky, 2004; Centola et al., 2000). In contrast, scripted curricula (e.g., Gobert et al., 2004; Horwitz, Gobert, Wilensky, & Dede, 2003) start out with well-defined content coverage. Whereas studies of the scripted curricula report positive learning results, they do not necessarily afford insights into areas outside of their target phenomena in their more "convergent" approaches, nor allot sufficient time for a deeper examination of elementary "under-the-hood" behaviors. Since these curricula do not make use of programming or modeling, it is understandable that students' familiarity with the behaviors and rules may not be as well-developed as in PBC modeling activities. In addition, since one key goal of MaterialSim is to train students to see commonalities across materials science phenomena, having a *strong programming and modeling component* was a key design principle.

Creating models is not foreign to undergraduate engineering – it is common for engineering students to have modeling assignments and learn several programming languages while obtaining their degree. However, traditional model-based activities in engineering oftentimes do not afford understanding of microscopic behaviors or elementary physical/chemical principles. Therefore, another key design principle is to *build activities which foreground these microbehaviors*, and in which students develop a high level of familiarity with the language and the ABM paradigm. In this study, the design foci are:

- *Programming exemplars* (solidification and grain growth) as to present students with the important algorithms and coding examples which could be useful in the process of building other models.
- *Support materials* to help students in learning how to program in NetLogo.
- *Easily transportable code examples*, which students could easily reuse across models.
- *Readily-available validation tools*, as to enable students to verify the validity of their models.
- A *persistent library* of student-generated models from previous years, from which students can reuse code and get inspiration for their models.

MaterialSim's grain growth model, the main exemplar model of the suite (Fig. 2.4), was conceived to enable four kinds of activities:

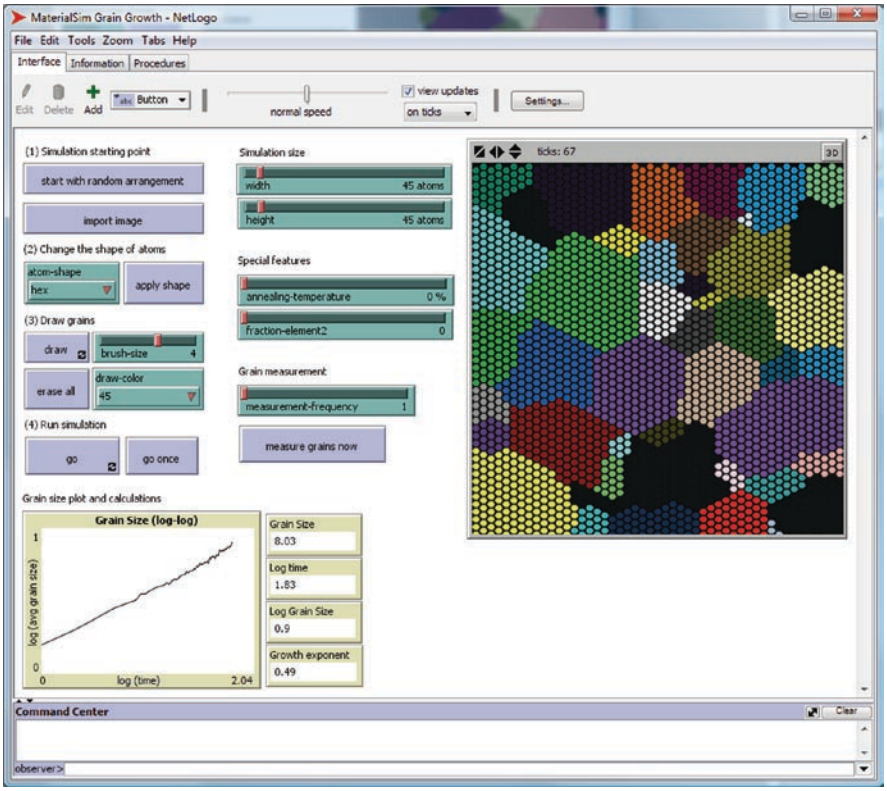
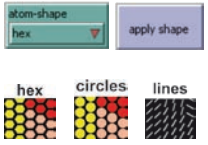
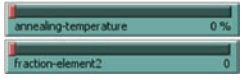


Fig. 2.4 MaterialSim's grain growth model

- *One-dimensional exploration*: Users can change variables, draw microstructures using the cursor, and observe their behavior over time.
- *Multidimensional exploration*: Students can run experiments sweeping entire parameter spaces, to determine critical points, system rules, mathematical relationships, and patterns.
- *Bifocal exploration* (Blikstein & Wilensky, 2006b): Students can connect real-world and virtual experiments, importing digital pictures from real experiments, and observing their "virtual" evolution. "Bifocal" refers to the simultaneous focus on the model and on the physical phenomenon.
- *Model building*: Students can change, create, or extend the system by coding their own procedures, modifying existing ones, or creating whole new models from scratch, by using the NetLogo modeling language.

In addition, the grain growth model offers a number of learning-oriented features, summarized in Table 2.1

Table 2.1 Summary of the learning-oriented features of the grain growth model

Simulation setup	Atoms' shape	Drawing tools	Additional parameters
(1) Simulation starting point  (4) Run simulation 	(2) Change the shape of atoms 	(3) Draw grains 	Special features 
Users can start either from a random arrangement of atoms or from a preset stage, which can be drawn by the user using the mouse or converted from a digital photo. This enables easy exploration of "what-if" scenarios and the comparison with real-world data	The appearance of the atoms can be changed for better visualization of particular aspects of the phenomenon. The "lines" mode, for example, was useful to understand the differences in crystallographic orientations	To investigate particular scenarios, users can draw their own microstructures and observe their behaviors, or retouch existing microstructures	The "annealing-temperature" slider enables experiments with different levels of thermal activity. The "fraction-element2" slider introduces a percentage of dispersed particles of a different material in the sample. Therefore, users can change the temperature and the type of material in order to study their effects on grain growth

Research Design and Methods

The research took place during three spring quarters at the materials science department of a midwestern research university, with sophomore students enrolled in the "Microstructural Dynamics" undergraduate course. In the first year (2004), six undergraduate students (volunteers) participated in the study. In the second year (2005), 11 students volunteered to participate, and 4 students participated in the third year (2006), with 21 participants over 3 years. The average class size was 15 students. Each student participated in two individual sessions. The first, 75-minute long, was comprised of the following parts:

- Short Likert-scale/open-ended presurvey to assess students' familiarity with computers and their attitudes about the course.
- Preinterview about grain growth and related phenomena, in which students were asked the content questions during a semistructured interview. These questions were based on exams and assignments used in the course (for example, "What is a grain?" "What is grain growth?" "What is the driving force for grain growth?" "What is the effect on grain growth of dispersed precipitates?")
- General presentation of the NetLogo programming environment.
- Demonstration of five canonical agent-based models from the NetLogo models library (fire spread, virus contamination, racial segregation, gas molecules in a container, and a chemical reaction).
- Hands-on interaction with one MaterialSim model: grain growth (with simultaneous interview). This included running the model with different values for matrix size, temperature, composition, as well as recording and plotting experiments sweeping the whole parameter space of one variable.

As homework, participants were asked to choose a challenging and/or interesting topic from the course and think of a model to build, which would be implemented during the next session. Students also had the option of extending the functionality of the existing grain growth model.

The second session (150 minutes) was dedicated to:

- Introduction to the basic commands of the NetLogo modeling language.
- Implementation (i.e., coding) of the new model. Participants were always in front of the computer and in control of the task. The first author would help students as needed with language commands and general programming issues.
- Final interview.

We scheduled the sessions approximately one week after students' exposure to the topic of grain growth in their regular classes. All sessions were videotaped, and students' computer interactions were recorded using real-time continuous screen-capture software. Approximately 65 hours of video were captured, which were selectively transcribed and analyzed. Experiments conducted by

students, as well as the models they built, were saved and analyzed. The first author attended the Microstructural Dynamics course 2004, 2005, and 2006, and analyzed the class materials and related literature. The classroom observations also generated data about the number of equations, variables, drawings, and plots explained during the class periods (and time spent in each item). Finally, participants were asked to fill up an anonymous web-based postsurvey, as to assess their (self-reported) interest and motivation doing the study, as well as usefulness of computer simulation for understanding certain topics in Microstructural Dynamics.

Data Analysis

Preinterview Explanations

The preinterviews were semistructured, following the questions listed in the Research Design section. At times the interviewer would ask additional questions to further explore one aspect of the responses. Students could also draw pictures to illustrate their explanations, which were all scanned. It was an open-book interview so that students could use any class materials, books, or websites to answer the questions. For the analysis, we randomly selected six students from the first two studies (2004 and 2005). The goal of the preinterviews was to evaluate students' explanations of core content in materials science, to which they were exposed during their regular classes one or two weeks before the interview. For the analysis, we compared explanations for the same phenomenon across different students, and also parsed and coded each explanation as to understand in detail the materials science concepts present in each, as well as how they were connected. In what follows, we will summarize results which were more comprehensively analyzed elsewhere (see Blikstein, 2009, and Blikstein and Wilensky, 2009).

The data shows that even for basic topics, such as describing what a grain is, students explanations were surprisingly dissimilar. Students resorted to a variety of metaphors and models for characterizing a grain: Betty,[5] for example, based her explanation on the visual appearance of a grain seen under the microscope. Liz tried to base her explanation on the appearance of a grain of rice. Ken tried to explain grains using another topic in the course, dislocation theory, which deals with the atomic structure of a material. As the interview progressed and questions started to deal with more complex and dynamic phenomena, the diversity of explanations just increased.

[5] All names were changed for anonymity.

When explaining what grain growth was, Bob used the metaphor of free will ("molecules come into the grain and line up"), and employed ideas about diffusion and impurities in contradictory ways. He did not resort to the Laplace–Young equation to explain the process of decreasing free energy by simply increasing the curvature radius. He incorrectly associates excess free energy to impurities or imperfections in the crystal structure, taking purity as a synonym for low energy, and grain growth as a cleansing mechanism by which grain boundaries would "sweep" impurities out. However, the Laplace–Young equation (studied in class) states a very different idea. Namely, the driving force is the curvature or pressure difference – impurities are not eliminated by grain growth, and growth can exist in pure materials. Betty, when trying to answer the question, mistakes grain growth for another phenomenon, recrystallization, which was taught in a previous section of the course. In recrystallization, similarly, crystals grow, but the mechanism and the kinetics are quite different. Ken, departing from yet another idea (rules of thumb about curvature), stated that "curvature is not good, so they will want to shrink."

When asked about the effect of impurities on grain growth, again, students tried to employ a variety of micromodels: a *force-feedback* model, where impurities particles pull boundaries away, slowing grain growth; a *classical mechanics* model, in which grain boundaries can "hit" a particle and slow down, models based on atomic movement inside the material, or purely *geometrical* models, in which the shapes of grain would change with no impact on the behavior of the material. As an example of a prototypical response, let us observe an excerpt of Ken's interview:

Interviewer: What is the effect of dispersed particles?

Ken: if you have two precipitations and if you have a dislocation line, you need to exert a force Tau on it, to move the dislocation line, but once it gets to the precipitation, it has to bow out and that will cost more energy, so if you have precipitations it will strengthen the material and that depends on the density of precipitations.

Interviewer: So grain growth slows down or not?

Ken: That I am not very sure.

Ken knew how to recite back pieces of the theory (even mentioning the name of a variable, "a force Tau"), but could not articulate its physical significance, and failed to identify the overall effect of impurities in grain growth. Indeed, our classroom observations showed that instructors overloaded students with a multitude of equations and models without necessarily making a clear distinction between the atomic behaviors themselves and the mathematical descriptions of those microbehaviors. In the interview, the consequences of the myriad of fragmented pieces of information and models to which students were exposed during class were apparent. Students' explanations, sewn together on-the-fly, employed incomplete fragments of variety of models, erroneously blended different topics (recrystallization, dislocations, grain growth), and often mistakenly used the standard vocabulary and rules of thumb of the field. What is more,

none of the students (even considering the entire group of 21 students participating in the study) even tried to use the standard mathematical equations to explain the phenomena. The data suggests, therefore, that the "many-to-one" approach used in class had detrimental consequences for student learning, i.e., the representational infrastructure of aggregate equations was not a good match to the content.

First Session of the User Study: Introduction and Model Exploration

The first session was dedicated to the exploration of the grain growth model. The first activity was simple: observe and reflect on curvature as a driving force for grain growth. Most of the students knew, from their previous classroom instruction, that large grains "consume" small ones, growing toward their center of curvature, and high-curvature boundaries tend to disappear. However, those concepts appeared to be isolated ideas, separate phenomena, and hardly connected to the Laplace–Young equation, which was supposed to be the mathematical evidence for the aforementioned phenomenon.

The goal of this activity was twofold. First, assess students' preexisting understanding of the phenomenon. Secondly, we carefully observed the cognitive shift as the simulation progressed (Siegler & Crowley, 1991). This activity consisted in drawing two grains divided by a curved surface and observing their behavior. The pictures below are snapshots of the dynamic simulation that students observed (Fig. 2.5).

Before the simulation, most students were unsure of what would happen. Approximately half thought that the larger grain would grow at the expense of the smaller, regardless of the curvature of the boundary separating them, while the other half considered concavity, rather than size, as the main criterion. As they started the simulation, and saw grains growing toward their centers of curvature, they observed that the movement was not smooth or unidirectional, but that there was intense activity on both grains with random flipping of atoms.

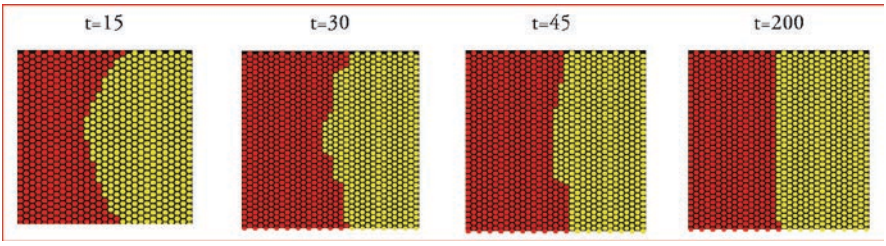


Fig. 2.5 The evolution of a curved grain boundary

The following excerpt suggests that visualizing this progression sparked some changes in Liz's understanding:

> **Interviewer:** Can you describe what you see?
>
> **Liz:** Just because one grain has a concave side and the other has a convex side, so it comes in towards the concave, because... [pause] does line tension applies in this situation?
>
> **Interviewer:** Line tension?
>
> **Liz:** That might be from dislocations... I might be mixing them up. Just because... when you have something... part of the grain is like, curving in, mostly likely other parts of the grain are curving in, so the tension of the grain boundary lines, so the force outside is greater than the force inside, so it will like shrink, it looks like that probably be like straight in the middle, rather than entirely red... just because if the red part also has some concave thing that is off the screen it will just like go together.

Liz is apparently mentioning ideas derived from the Laplace–Young equation, which relates surface tension and curvature. However, she cannot yet think at the "micro" level: To visualize what is happening on the computer screen, she has to imagine a large circle going off-screen, which is probably a consequence of what she remembers from class, where grains were always approximated as spheres. She does not yet construe the local interactions along the curved interface as a driving force, but only the "macro," aggregate-level effect of curvature.

The next activity was to draw a microstructure with many grains, but with one of them a lot smaller than the others, as we can see in Fig. 2.6.

Watching the evolution of this new microstructure was a crucial experience for Liz. She started to transition from memorized rules of thumb and topic-specific models to micro-level reasoning, which would lead her to generate hypothesis about the grain growth law by herself. This excerpt took place when she was observing a triple point – a region where three grains meet and the probability of an atom to flip to

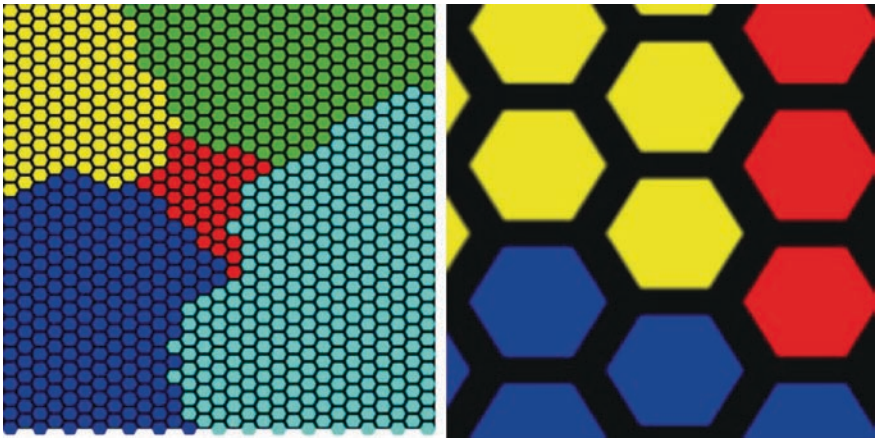


Fig. 2.6 Four large grains surround a small grain (*left*), and a zoomed-in view of the structure showing a triple point (*right*)

any of the surrounding grains is the same, as there are *two* atoms of each grain around the central element (see Fig. 2.6.) While observing this phenomenon, Liz was told to zoom in and out the simulation, to also see what was happening at the microlevel (following a single atom).

Liz: Right here there is an equal position for red, yellow, and blue, but it just happens to be that blue won, it keeps winning.

Interviewer: How would you explain that?

Liz: Because... if you look at one of those points, either of the three colors, they all have the same number of other colors around it, so it is not favorable to choose one or the other...

Interviewer: What angle is here?

Liz: Oh, so this is the 120-degree angle between the... [pause]

Interviewer: Did you talk about it in class?

Liz: Briefly. He [the professor] said that when you reach a triple junction, it will become 120 degrees.

Interviewer: So are you saying that there is an equal probability?

Liz: Well, I just don't understand why blue is doing so much better, in general. Eventually just one has to become bigger, because this is the most energetically favorable thing, so maybe... blue was bigger, but now yellow is coming back, so maybe next time blue gets bigger again, and they will just keep going. Maybe it will just be like that for a long time.

Interviewer: So what happens to growth speed?

Liz: Eventually they will get like... two big ones... and then it will take forever.

Interviewer: So what could be the law?

Liz: It will eventually taper off... to some point... because if you have a lot or grains then you will... the rate of increase will be faster, but when average grain size increases it gets harder and harder to increase the rest of them, so it just goes...

Interviewer: Why is it harder and harder?

Liz: Just because there isn't a distinct... [pause] being in this orientation is more favorable than this other one so you have to pick and choose... the grains are doing that, but it is not happening quickly just because you know, either one can happen.

In this very short time watching the model, Liz was able to understand and generate hypotheses about two essential ideas: triple points and logarithmic laws (the literature refers to these ideas as particularly hard to understand (e.g., Krause & Tasooji, 2007)). Rather than trying to assemble statements pulled from regular instruction, Liz departed from what she knew about other phenomena and what she was actually seeing in the computer model. Even without formally mathematizing the time dependency of grain growth, she understood the reason for the triple point to be considered a "low-mobility" point in a microstructure. The central atom has two atoms (out of six) of each of the surrounding grains as neighbors, so the switch probability is the same (1/3), and there is no preferred growth direction. She also realized that the time law would not be linear: growth speed decreases over time and eventually "tapers off." Rather than *being told*, Liz arrived at this conclusion *on her own*, by drawing microstructures, changing variables, and observing the dynamics of the

simulation. Particularly, when asked about the fundamental reason for the "tapering off" of grain growth, she affirmed that "[…] because there isn't a distinct orientation [which] is more favorable" – in other words, she got at the core of the explanation, the fundamental atomistic principle. This same idea could be useful to explain other phenomena in materials science, and we will see how students applied such generative ideas to other phenomena in the next section.

Similarly, Peter and Elise, notwithstanding their initial difficulties in explaining grain growth during the preinterviews, understood the logarithmic nature of the grain growth law:

Interviewer: What could be the rule for grain growth speed?

Peter: As the grains get bigger, each grain is increasingly hard to take away from because it's bigger, so the interfaces start to be between two large grains, instead of small grains, so an interface between a large grain and a large grain is less likely to have a lot of movement because both grains are large and they are already in a state where they don't want to shrink.

Interviewer: What will happen to this surface [between two grains]?

Elise: [It'll] shift up to be vertical. [Looking at the computer model.] Yes, it's just getting flatter.

Interviewer: Why do you think it wants to get flat?

Elise: It's like the nearest-neighbor thing, these want the most nearest green neighbors, the red ones want the most red ones.

Interviewer: [some minutes later, she is looking a triple point] What's happening here?

Elise: It has the same number around each other, so, the red, the angles are all at equilibrium, they are all a stable formation.

Interviewer: And what's that angle there?

Elise: It's a hexagon, I guess it's 360 divided by three, 120.

Generally, most students knew that the small red grain in Fig. 2.6 was going to disappear. From their reactions while observing the simulation, they seemed to be expecting a unidirectional "animation" of the grain being "eaten" by the surrounding ones, and eventually disappearing. This was consistent both with the heuristics and the types of results of aggregate tools, animations, and equations commonly seen in class, which are processes that happen unidirectionally and deterministically. However, what students observed was different: behaviors emerging from local interactions, which take place with some degree of randomness. At times, the small grain would grow, but most of the times it would shrink. Some of the students wanted to slow down the simulation and use the "zoom" tool to see the process in more detail, which meant they could only see the microlevel phenomenon (atoms flipping to different orientations). By zooming out again, they could observe the *emergent* behavior: curved surfaces disappearing as the Laplace–Young equation would predict. Thus, there is a qualitative difference between traditional learning tools and agent-based modeling: not only are students observing an expected outcome, but also *they are able to see the process unfolding* at various levels. The simulation was visually similar to the phenomenon, but, most importantly, its algorithm loyally emulates the micro-level processes underlying it. This is different from purely numeric

simulations in which students are able to compare only outputs, and not the processes as they unfold. In addition, words commonly used in the classroom, such as "shrink," "consume," and "growth" acquired a new meaning. Those metaphorical terms, as our preinterview data suggested, can mislead students to interpret literally their meaning – working with MaterialSim, students realized that grains were not actually being "consumed" or shrinking: atoms were just switching places, and the metaphors were just describing the net, aggregate effect of such behavior. This was a central element of the whole experience and, as we will observe, deepened as students progressed in the study.

The last activity of the first day was to run automated experiments using NetLogo's "BehaviorSpace" module. This NetLogo feature enables users to automatically run hundreds of simulations, each under different parameter settings, sweeping entire parameter spaces. Students ran at least one set of experiments, plotted the data, and came up with theories to describe the phenomenon. Most students chose to model the influence of dispersed impurities on grain growth. The textbook explanation of this phenomenon takes approximately four pages. It begins with an account of how a force *P* appears when a grain boundary attempts to go through a particle, and then calculates the drag force by means of geometrical approximations (see Fig. 2.7).

Departing from those geometrical approximations (for example, all particles are considered to be perfect spheres), the formula is obtained with a series of

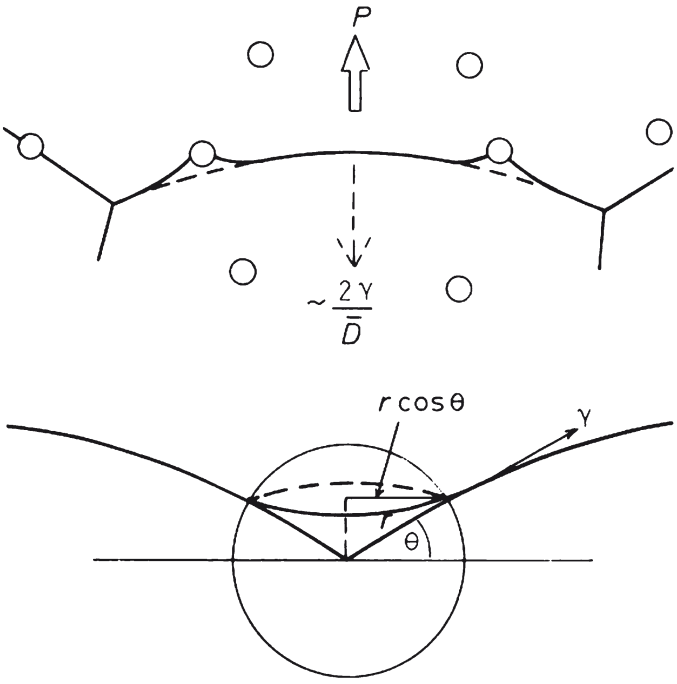


Fig. 2.7 The textbook picture explaining how dispersed particles affect boundary migration (Porter & Easterling, 1992, pp. 141)

derivations (Porter & Easterling, 1992, pp. 141), which relates the fraction of dispersed particles (f), the mean radius of the particles (r), and the maximum particle size after grain growth (D_{max}):

$$P = \frac{3f}{2\pi r^2} \cdot \pi r\gamma = \frac{3f\gamma}{2r} \Rightarrow \frac{2\gamma}{\bar{D}} = \frac{3f\gamma}{2r} \Rightarrow \bar{D}_{max} = \frac{4r}{3f}$$

However, the NetLogo algorithm is not based on this formula, or its geometrical approximations. Before running his experiments, Bob was asked if the model could actually match the "official" equation, since they departed from very different ideas, and he was skeptical. Thus he programmed NetLogo to run a series of simulations with percentages of particles varying from 0 to 8% (see screenshots and individual plots of grain growth speed in Fig. 2.8). He also constructed a plot to aggregate the results across all experiments, and subsequently tried to compare their own curve with the theoretical data (dotted line in Fig. 2.8's lower plot). To his surprise, the two curves had a very reasonable match. Other students, with slight variations, undertook the same project, or selected different aspects of the phenomenon. By exploring entire parameter spaces, and having not only the dynamic visualization but also actual numerical data to base their explanations on, these students were able to further generate original hypotheses and find meaningful correlations.

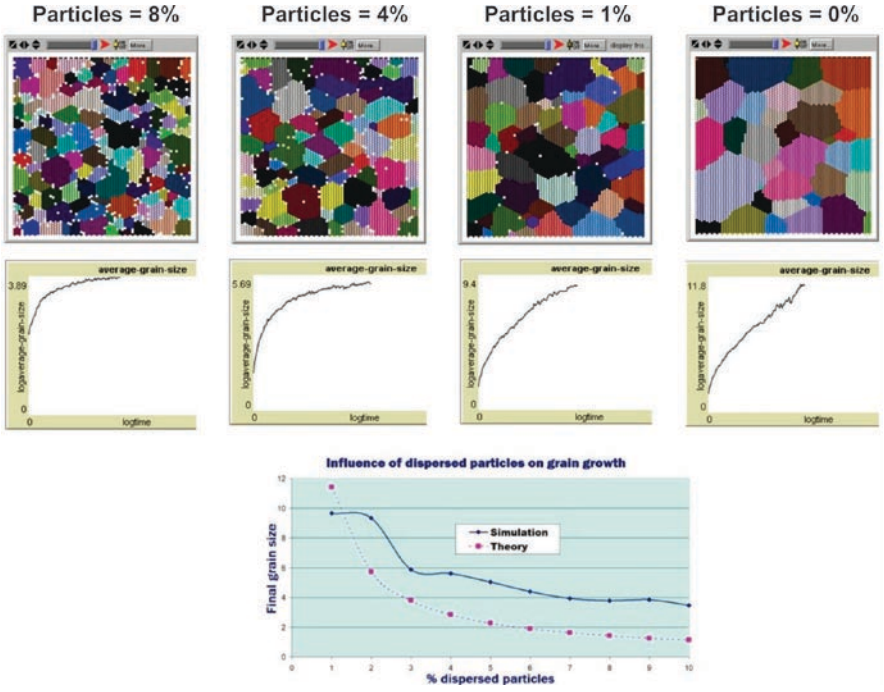


Fig. 2.8 Sequence of screenshots from Bob's experiment

Second Session: Building Their Own Models

The second session was the model construction part of the study. Students had 2½ hours to learn the basics of the NetLogo language and program a new model of a materials science phenomenon. For this session, which took place 2–3 days after the first session, students were asked to bring one idea of their own for a new model. They pursued questions of their interest or problems that they did not understand during regular instruction. By authoring new models or new features for the existing models, they could elaborate on answers to their research questions. Student achievement was impressive. A comparison between the preinterview data, when students relied on ready-made statements about the phenomena, and their performance on the last day of the study, when they built their own models relying just on fundamental atomic behaviors, suggests that student contact with an agent-based environment effected conceptual gain. Even more than exploring the existing models, constructing their own models was a transformative experience for most. In this section we will narrate and analyze some of those learning trajectories. The models chosen for this analysis represent typical student work, and the particular choice of which students to include in the data analysis attempted to provide representative examples of the various affordances of ABM employed by students.

Betty's Model

Betty built a model to investigate grain growth with a new and important feature: taking into consideration the misalignment between grains (see Fig. 2.9). In her innovative model, the more misalignment across the boundary of two grains, the harder it would be for an atom to jump from one grain to another. The construction of this model presented Betty with many challenges. The first was to convert the grain orientation's angle, which could lie in any of the four quadrants, to a normalized quadrant-independent measure that would be easier to compute. Betty's solution,

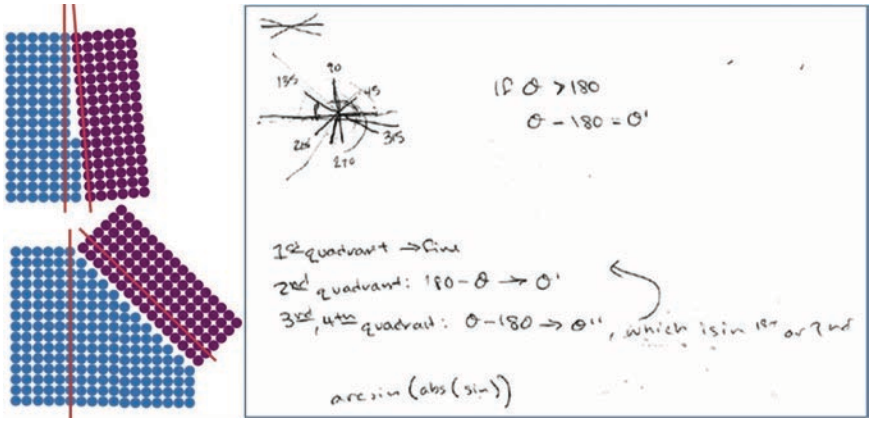


Fig. 2.9 Betty's sketches about angles, sine and arcsine

after much thinking, sketching, and trying out different trigonometric functions, was to use the *arcsine* function. The following picture shows some of her reasoning. From her drawing, we can observe that she was using geometrical analysis from a "micro" level, examining individual atoms, and trying to design an algorithm to account for the trigonometric issues in calculating their misorientation.

She considered that the probability for an atom to jump to the next grain should be dependent not only on the number of different atoms around it, but also on the average misorientation between the two grains. Low misorientation would promote easier migration, thus she needed a function to calculate the misorientation, and then to add a misorientation factor to the previous grain growth algorithm. Apart from the initial difficulty in figuring out the best trigonometric function for the angle comparison, Betty knew what she needed to do, without resorting to any of the textbook formulas. For her, at the microlevel, adding the misorientation effect was quite easy.[6] Therefore, she simply added one command to the original grain growth model to implement her change. Previously, in the traditional aggregate equation-based approach, making such a change would require long and mathematically demanding derivations. The resulting code of her misorientation calculating function was:

```
;;calculates the absolute value of the arcsin[7]
to-report misorientation [angle]
  report asin (abs (sin (angle)))
end
;;calculates the absolute value of the sum of the two
arcsins
to-report calculate-misorientation [angle1 angle2]
  report abs (misorientation (angle1) + misorientation
(angle2))
end
;;reports the average misorientation for a given atom
to-report compare-misorientation

let i 0

ask neighbors6
     [
     ;;calculates the misorientation between the current
     atom and each of its 6 neighbors
     set total-misorientation (total-misorientation +
     calculate-misorientation heading (heading-      of
     neighbors6))
     set i i + 1 ;update counter
    ]
```

[6] On a more advanced level, similar research was undertook and published by researchers, such as Ono, Kimura and Watanabe (1999)

[7] In the Netlogo programming language, semicolons mark the start of a comment line. Programmers use comments to clarify and annotate their code.

```
;;returns the average misorientation
report (total-misorientation / i)
```

end

Then, having her reporter agents calculate how much each given atom would differ from its neighbors angle-wise, she changed the original grain growth procedure, adding one extra simple "if" statement:

```
;;OLD PROCEDURE
  if future-free-energy <= present-free-energy
      [set heading (future-heading)]

;;BETTY'S NEW PROCEDURE
  if future-free-energy <= present-free-energy
        [
           if (present-heading - ([heading] of one-of
             neighbors6) < misorientation)
                  [set heading (future-heading)]
        ]
```

Yet, aggregate and macroscopic models do not afford such insights as well. The agent-based approach, conversely, provided a "low-threshold" entry point for Betty to implement her ideas by constructing models. Her model was very consistent with known theory, even though she was not cognizant of this theory prior to the interventional study. Betty's model illustrates one of the main affordances of the agent-based representation: at the micro level, *the mathematical machinery required to add new phenomena or parameters to an existing algorithm is much simpler* than in traditional representations. Instead of employing numerous equations to add her misorientation effect, just a few lines of code, at the microlevel, were sufficient.

Jim's Model: Polymer Chains

Jim was taking a polymer science course at the time, and in previous classes he had learned about polymer chains and how they moved. Polymer chains can move and expand, but in most cases not if that process ends up breaking the chain itself. When he was choosing the idea for his authored model, he very quickly realized that the neighborhood based grain growth algorithm could be a good start for a polymer model. Very quickly Jim computationally modeled atoms for his polymer chains in this way:

Atoms are:

Moving randomly in all four directions (0, 90, 180, and 270 degrees)
But
Not breaking the chain
Not crossing the chain

His NetLogo implementation followed these three simple steps.

```
to move
  ;; choose a heading, and before moving the atom
  (monomer),
  ;; checks if the move would break or cross the chain
  set heading 90 * random 4
  if not breaking-chain? and not crossing-chain?
     [forward 1]
end
```

To check if the monomer movement would break the chain, he wrote a procedure that searched atoms at the four orthogonal directions. In case there were any atoms there, the procedure returned "false" and that atom did not move. A similar reporter was done for crossing chain, but with a different set of neighborhood points. The model's interface (and typical initial setup) can be seen in Fig. 2.10.

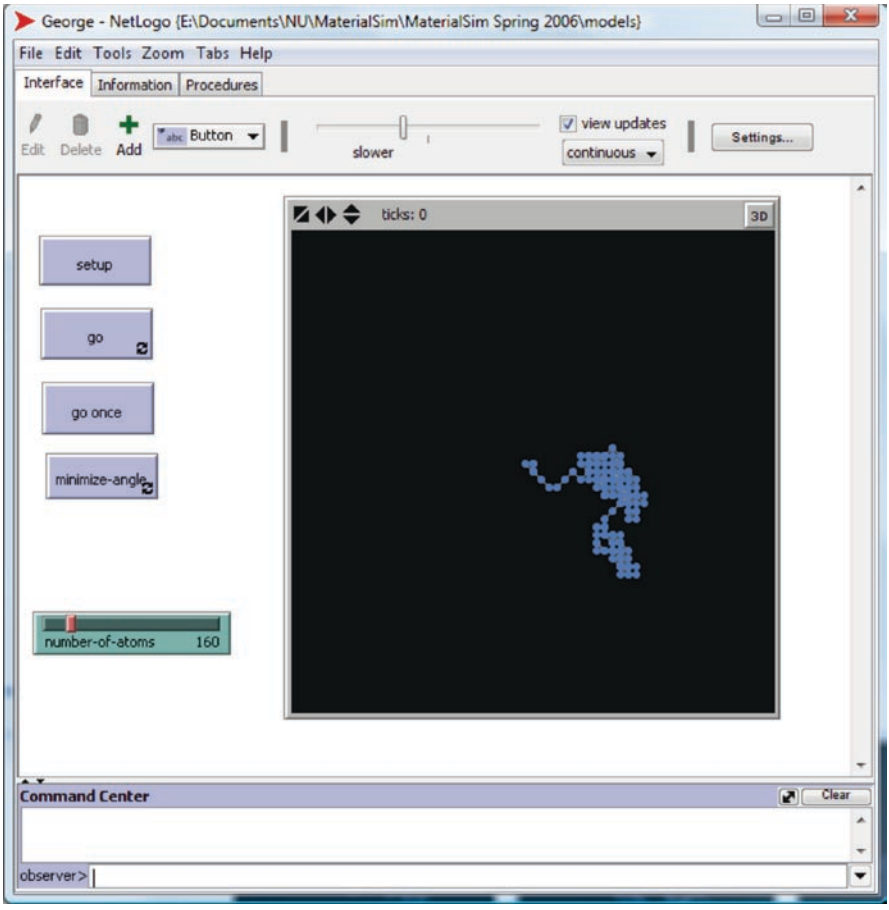


Fig. 2.10 Jim's "Polymer Chains" model

But the model worked only in a very limited way, because if there were too many atoms "clumped" together, they would never get a chance to move, since any movement would either break or cross the chain. One reason for this problem is that it failed to incorporate the attractive and repulsive forces between atoms. Therefore, even though some atoms could move, the chain would not greatly expand because most atoms were within a "one" radius of each other. Jim needed a spring-like repulsive force activated at particular time steps to relax the system, and an attractive force to keep the atoms close to each other. His answer was to create the one extra procedure with just a single line of code using NetLogo's *layout-spring* command, which applies to spring-like force to the links between the agents:

```
;; makes links act as springs (the number after the com-
mands are parameters of the spring)
layout-spring atoms links .4 1 0.1
```

By blending two algorithms, Jim got his model to work exactly as the animation shown in class by the professor – but in a short program of about 15 lines of code. In Fig. 2.11 we have a typical evolution of a polymeric chain. On the last frame (bottom, right), the atomic "movement" procedure was turned off, so just the spring-like forces were in place, generating a smoother chain.

Jim's model is another example of two important affordances of ABM: the easy *blending of algorithms* (in this case, he was able to easily "blend" two typical ABM

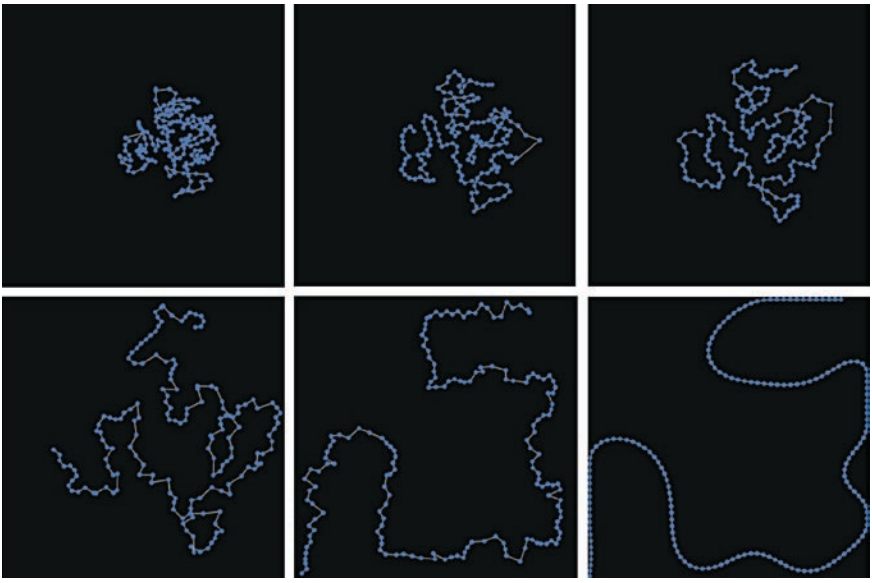


Fig. 2.11 The evolution of a polymer chain in Jim's model

algorithms, spring-behavior and restricted movement), and the *one-to-many generativity* (again, similarly to grain growth, a neighborhood-checking mechanism was at the core of the model). After understanding in detail the principles and algorithm behind the grain growth model, he was able to identify other opportunities to employ the same principles to model and understand other seemingly unrelated phenomena.

Peter's Model

Peter's model was another example of the one-to-many, transferable affordance of the agent-based representation. In the pre-survey, he identified diffusion and interface-controlled reactions as some of the hardest topics in the course. In the second session, he chose these topics for building a model. In materials science, it is particularly important to distinguish transformations that are interface-controlled (i.e., the slowest phase happens at the interface of the two materials) from diffusion-controlled (the slowest part is for atoms to "travel" to the interface, where the actual reaction is fast). Knowing the slowest phase of a given process, engineers can greatly optimize it. Peter's purpose was to build a model to investigate this phenomenon. Its textbook explanation is a five-page narrative with five different equations, which are put together to show that the driving force (referred to as $\Delta\mu_B^i$) is proportional to the temperature and the difference in concentration:

$$\Delta\mu_B^i = \mathrm{R}T\ln\frac{X_i}{X_e} \cong \frac{\mathrm{R}T}{X_e}\left(X_i - X_e\right)$$

Where X_i and X_e are the chemical compositions of the two materials, T is the temperature, and R is the gas constant (Porter & Easterling, 1992, pp. 177).

Peter ignored the existence of this long sequence of equations. He started his model from scratch, and his first step was to identify the basic atomic behaviors needed for implementing his idea. In his model, there were two types of materials in liquid form, and one type of solid material. Therefore, he needed one mechanism for atoms in the liquid to move, and one mechanism for liquid atoms to become solid. He concentrated in the microrules concerning the phenomenon, and realized that the rules he needed were not very different from the rules present in other models. After all, liquid atoms were just moving randomly and "bumping" into a solid, sticking to it according to a certain probability. The Solid Diffusion model, present in NetLogo's models library, had an efficient algorithm for making atoms move around in a material. The Grain Growth model provided Peter with the idea for the liquid-to-solid transformation. Even though those two models (Solid Diffusion and Grain Growth) had significant differences compared to the model Peter wanted to build, he managed to identify the common useful microrules, copy the code from one model to the other and, very importantly, make the necessary adaptations.

Peter's algorithm was straightforward: if the atoms are in the liquid, and they bump into a solid, they become solid (with a certain probability, dependent on their chemical properties), hence:

```
if ((breed = element1-liquid) and ;;if you are an atom
in the liquid
    (neighbor-breed = solid) and ;; and one neighbor of
yours is in the solid
     (element1-probability > random-float 100)) ;; and
depending on your diffusion speed
        [
          set color neighbor-color ;; switch the atom's
          color
          set breed neighbor-breed ;; switch the atom's
          breed
          ]
```

If the atom is in the liquid (breed different than solid, or "!=solid" in NetLogo language), and it meets an atom different than itself, the atoms switch places – in other words, diffusion is taking place:

```
if ((breed != solid) and ;;if you are in the liquid
     (neighbor-breed != solid) and  ;;and one neighbor
     of yours is also in the liquid
     (diffusion-speed > random-float 100)) ;; and depending
     on your diffusion speed
       [
         set [color] of random-neighbor color ;;switch
           the neighbor's color
         set [breed] of random-neighbor breed ;;switch
           the neighbor's breeds
         set color neighbor-color ;; switch the atom's
           color
         set breed neighbor-breed ;; switch the atom's
           breed
       ]
```

Note that the idea of asking atoms to check their near neighborhood came from the Grain Growth model, whereas the idea of atoms switching places as a way to diffuse through a material came from the Solid Diffusion model. In two dozen lines of code, and less than 2 hours, Peter was able to model both diffusion and solidification, manipulating exclusively local rules, and had a model the complexity of which was far beyond what is expected from the Microstructural Dynamics course, considering the classroom observations and analysis of class materials. Nevertheless, just as other students, he was concerned with the correctness of his work. He generated a series of plots and screenshots to match his data with the textbook plots, part of which are shown in Fig. 2.12.

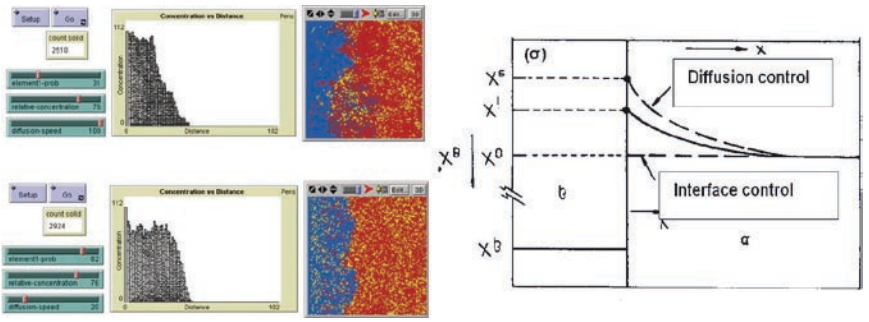


Fig. 2.12 Results of Peter's model with diffusion control (*top*, with diffusion speed = 100), interface control (*bottom*, with diffusion speed = 20), and the chart from the textbook, where we can identify a similar same shape for the two concentration curves. (Note that this last chart was rotated for clarity)

At the end of the session, the first author asked Peter about the next steps in his model's implementation, had he more time to work on it. Again, he demonstrated a solid understanding of how to easily manipulate those basic rules to generate new models, for example, how to invert a process by simply changing the probability of its microevents:

> **Peter:** I did a liquid to solid model, now I want to be able to invert it, do a solid to liquid algorithm.
>
> **Interviewer:** And how would you implement it?
>
> **Peter:** It's simple: I'll just invert the probability. It's just the opposite probability. I don't have to change much.
>
> […]
>
> **Interviewer:** And how would you implement, for example, dispersed particles in a grain growth model?
>
> **Peter:** I would put in molecules that have zero probability to change to anything else, and zero probability of another molecule to change into them.

Peter's response demonstrated a deep understanding of the process and was in great contrast with his preinterview data, in which although he correctly identified and explained some phenomena, he failed to see how those principles and knowledge could be put to use to further his own knowledge about a particular topic or other phenomena.

Discussion

Computer modeling is posing a serious challenge to extant knowledge encoding schemes in engineering and materials science. Researchers have already detected this trend – computer modeling in materials science has more than doubled in the

last 10 years (Thornton & Asta, 2005). However, materials science students are still obliged to master hundreds of equations and isolated facts. Even if students were to somehow try to connect those equations into a coherent corpus, the mathematical machinery required to accomplish that would be too demanding for most to succeed.

The examples of students' model building we have described were implemented in less than 3 hours, *including* the time dedicated to learning the basics of the NetLogo language. The relative ease with which students developed their own models, even within such a short timeframe, shows that model building is an approachable task for undergraduate students and supports one of our main claims: agent-based modeling, for some fields of engineering, offers a more principled understanding of the natural phenomena, which, in turn, grants more autonomy for students in learning new content or deriving new theories on their own. Participant students had previous knowledge of the phenomenon from their class work. Nevertheless, during the preinterview, they demonstrated difficulty in explaining related phenomena in a coherent fashion, resorting to a range of fragmented models and metaphors. The implementation of their own model within an agent-based modeling environment provided students with fewer, simpler rules that were closely related to the physical phenomenon, thus enabling them to better understand and extend the model by adding new proximal rules for the agents.

We compiled evidence suggesting that the agent-based encoding is a good fit to content in materials science. First, the undergraduate courses are overloaded with highly specialized information. Secondly, students demonstrated difficulty in explaining even the most basic concepts in the field, with frequent "slippage" between levels. Thirdly, throughout the classrooms observations and the interviews, one striking revelation was that the agent-based approach was not a total unknown for textbook authors, teachers, and students. The textbook oftentimes makes use of microbehaviors, simple rules, and agent-based heuristics. When explaining grain growth, the textbook authors use an agent-based approach:

> […] A similar effect occurs in metal grains. In this case the atoms in the shrinking grain detach themselves from the lattice on the high pressure side of the boundary and relocate themselves on a lattice site on the growing grain. (Porter & Easterling, 1992)

However, the agent-based representation was in this context a mere illustration of the "real" content that would come after, encoded as equations. Arguably, even though the agent-based representations could be easier for students to understand, there was no technological infrastructure to "run" those models – the activities and software that we developed could provide this infrastructure. The availability of an expressive tool and an empowering learning environment were crucial elements. As a computational tool, NetLogo and its agent-based approach was a good fit for capturing students' intuitions and ideas at the appropriate level. In addition, the constructionist nature of students' interaction with the tool enabled them to build fluency with this new tool, and perceive themselves as scientists in their own right, transforming seemingly simple ideas and local rules into powerful kernels for scientific modeling. To further understand the cognitive model which

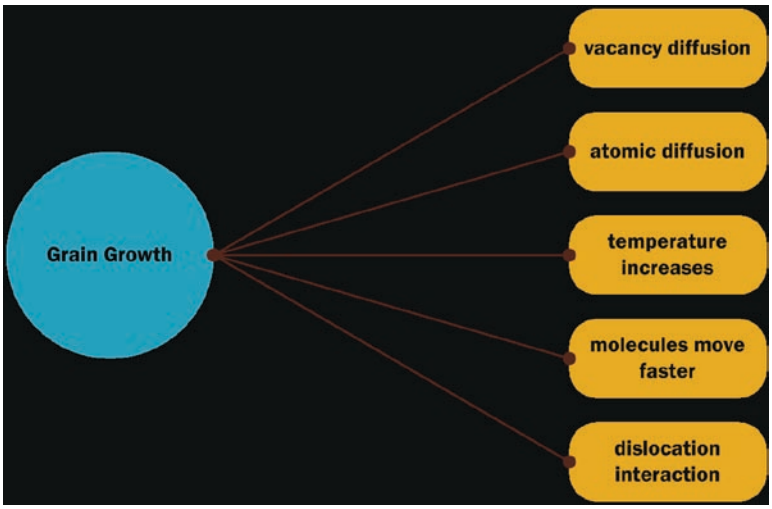


Fig. 2.13 Bob's one-level explanation

the ABM perspective might foster, let us consider again, for example, Bob's explanation of grain growth:

> **Bob:** Well, grains grow through diffusion, through **vacancy diffusion**, and **atomic diffusion**, for one, it is all over the place, **temperature increases**, **molecules move around** faster […].

His statement reveals a one-level description of the phenomena, which is compatible with our analysis of the current sparse and linear encoding of knowledge in materials science. Ideas such as "vacancy diffusion" and "increase of temperature" are connected to "grain growth" without a clear hierarchy (Fig. 2.13).

During the work with MaterialSim, students developed an extra "organizing" layer which grouped some of the surface manifestations on the phenomena under one unifying principle[8] (Fig. 2.14). Let us observe Liz's statement:

> **Liz**: It is because, it wants to be more **energetically stable**, or have **less energy** in the crystal, so it will grow, just to form one big grain, because that's the least energy configuration […]

Liz identified one unifying principle, "lowering free energy," from which many of those external manifestations derive. An agent-based modeling environment offers low-threshold tools to code and formalize this principle algorithmically, enabling her to "mobilize" this idea that was previously just a vague mental model. Finally, after the model building, students were able to mobilize these generalizable principles, encoded as computer code, to explain other phenomena that shared the same mechanism (Fig. 2.15).

[8]For elaboration on the idea of organizing layers, see "Papert's principle" in Minsky's Society of Mind (1996).

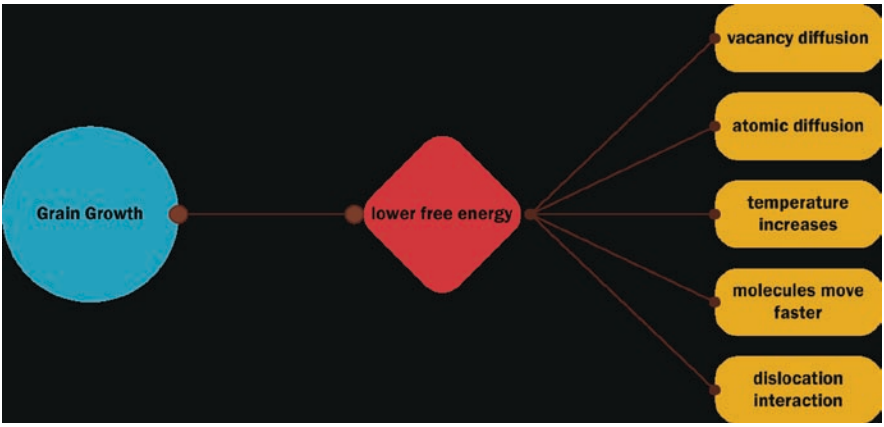


Fig. 2.14 Liz's two-level structure

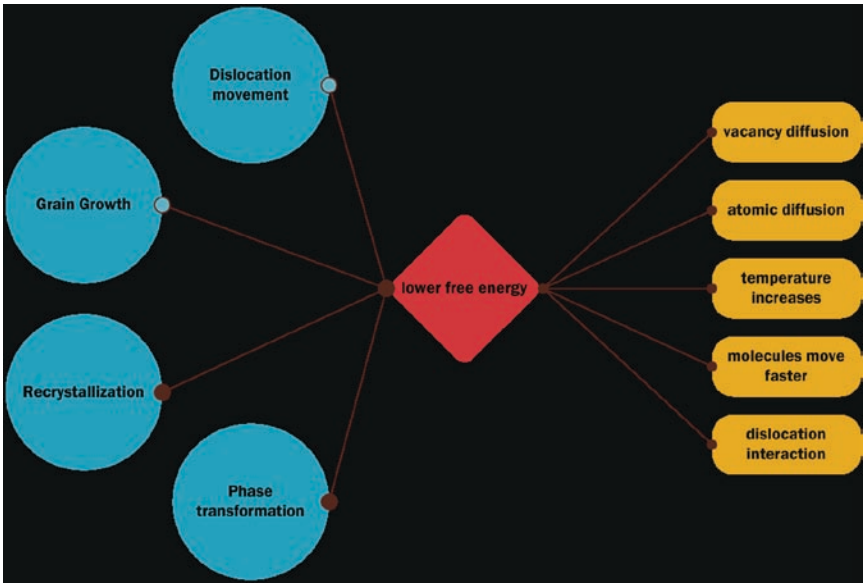


Fig. 2.15 A two-level structure with multiple phenomena

Conclusion

Design courses have become fashionable in many engineering schools. Robotics and engineering design competitions are common in various universities and even high schools. One question explored in this study was: can we extend the powerful

ideas about engineering design, constructionism, and project-based approaches to theoretical engineering courses in fields such as materials science, in which students' projects would be quite different from robots?

Rich, motivating learning is often achieved through an approach of *learning-by-doing*. In areas such as mechanical engineering, *doing* and *understanding* could be tightly connected. When students are building a gearing system, all the components are clearly visible. In areas such as chemistry, atmospheric science, biology, and materials science, this is not usually the case. Learners might observe overall effects while having little understanding of the underlying causality or the fundamental components of the system, since the actual phenomenon it too removed from human size or time scale. Moreover, teaching tools in those disciplines often have relied on "aggregate," formulaic descriptions. This study suggests that the fragmentation and opaqueness of such descriptions could constitute an obstacle to learning. First, the traditional aggregate equational descriptions are more phenomenon and context-specific, and do not enable students to make broader inferences about phenomena with similar micromechanisms. The mathematical machinery required to manipulate and combine aggregate equations is highly complex and constitutes an obstacle for many students. Second, these descriptions often lead to heuristics that generate overgeneralizations – students often had memorized ideas about phenomena in materials science about which they had no deep understanding. Third, the traditional descriptions are formally detached from representations of the actual physical phenomena, i.e., the aggregate formulaic descriptions don't inform students, at first glance, much about the atomistic mechanism of the phenomenon under scrutiny.

On the other hand, agent-based modeling seems to provide phenomenally isomorphic representations that can lead to deep conceptual insights about the content areas discussed in this chapter, for three reasons:

1. Students' experience interacting with the MaterialSim models, and building their own models, foregrounded the fundamental physical processes taking place in the material, namely atomic movement and free-energy minimization. Not only were most of the algorithms exclusively based on those processes, but also the design of the visualization schemes enabled students to *see* them unfolding in real-time. Students observed both favorable and unfavorable atomic jumps, grains growing and shrinking, expected and unexpected results.
2. A core feature of this design is that students can apply a small number of kernel models to capture fundamental causal structures underlying behaviors in apparently disparate phenomena within a domain. For example, a free-energy minimization algorithm could enable students to understand not only grain growth, but also annealing, interfacial energy, recrystallization, diffusion, and phase transformations, which are traditionally taught as *separate* topics with their own models and equations. Most students were able to create their own models by transferring some "kernel" algorithms from one model to another, making the needed adaptations.
3. One of the pillars of constructionist theory is the importance of students conducting personally meaningful projects. Even though students had significant insights about the phenomena by interacting with predesigned models, our data

suggest that coding their *own* models was a particularly valuable learning experience. It was during the model-building process – writing code, testing and debugging their theories, and reconciling them with previous knowledge – that students had a deeper and more intense exposure to the tools and methods of agent-based modeling, being able to develop enough fluency with the computational representations. In addition, we have shown that learning a low threshold programming language such as NetLogo and coding a model can be done in short enough time to be feasible in actual university classrooms.

In conclusion, the research reported here suggests that *less is more*. Specifically, our findings suggest that agent-based approaches to representing knowledge offer a radically different avenue for students to engage in scientific inquiry. Exploring and learning about just a few simple underlying rules of natural phenomena, given the availability of a computational medium to manipulate, represent, combine, and analyze them, appears to be more generative for students than the current teaching approaches in materials science and engineering that employ numerous aggregate, equation-based representations. We hope these findings inform future research and development in STEM education in so far as extending to theoretical science and engineering courses the principles of student-centered, constructionist pedagogies – in particular, using the tools and approaches derived from the complexity sciences and agent-based modeling.

Acknowledgments We thank Professor Tschiptschin (Engineering School of the University of São Paulo, Brazil) for his inspiration and support, Professor Srolovitz (Yeshiva University), and the members of the Center for Connected Learning and Computer-Based Modeling at Northwestern University. We also thank Ann McKenna from the McCormick School of Engineering at Northwestern University. Special thanks for Spiro Maroulis and Michelle Wilkerson-Jerde (Northwestern University) and Professor Dor Abrahamson (UC Berkeley) for invaluable comments on earlier drafts of this chapter. This work was funded under the umbrella of three NSF grants: REC-0126227, HCC-0713619, and NSF EEC-0648316.

References

Abrahamson, D., & Wilensky, U. (2005). *ProbLab goes to school: Design, teaching, and learning of probability with multi-agent interactive computer models.* Paper presented at the Fourth Conference of the European Society for Research in Mathematics Education, San Feliu de Guixols, Spain

American Society for Engineering Education. (2007). ASEE profiles of engineering and engineering technology colleges. Retrieved from http://www.asee.org/publications/profiles/upload/2007ProfileEng.pdf

Anderson, M. P., Srolovitz, D. J., Grest, G. S., & Sahni, P. S. (1984). Computer simulation of grain growth: I. Kinetics. *Acta Metallurgica, 32*, 783–792.

Berland, M., & Wilensky, U. (2004, April 12–16). *Virtual robotics in a collaborative constructionist learning environment.* Paper presented at the Annual Meeting of the American Educationalt Research Association, San Diego, CA

Berland, M., & Wilensky, U. (2005, April 11–15). *Complex play systems -- Results from a classroom implementation of VBOT.* In W. Stroup, U. Wilensky (Chairs), & C. D. Lee (Discussant),

Patterns in group learning with next-generation network technology. Paper presented at the Annual Meeting of the American Educational Research Association, Montreal, Canada

Blikstein, P. (2009). *An atom is known by the company it keeps: Content, representation and pedagogy within the epistemic revolution of the complexity sciences.* Unpublished doctoral dissertation, Northwestern University, Evanston, IL

Blikstein, P., & Tschiptschin, A. P. (1999). Monte Carlo simulation of grain growth. *Materials Research, 2*(3), 133–137.

Blikstein, P., & Wilensky, U. (2004a). *MaterialSim: an agent-based simulation toolkit for learning materials science*. In M. Hoit & T. Anderson (Eds.), *Proceedings of the International Conference on Engineering Education, Gainesville, Florida, USA*

Blikstein, P., & Wilensky, U. (2005a). Less is more: Agent-based simulation as a powerful learning tool in materials science. *Proceedings of the IV International Joint Conference on Autonomous Agents and Multi-agent Systems (AAMAS 2005), Utrecht, Holland.*

Blikstein, P., & Wilensky, U. (2006a). *From inert to generative modeling: two case studies of multiagent-based simulation in undergraduate engineering education*. Paper presented at the Annual Meeting of the American Educational Research Association, San Francisco, CA

Blikstein, P., & Wilensky, U. (2006b). 'Hybrid modeling': Advanced scientific investigation linking computer models and real-world sensing. *Proceedings of the Seventh International Conference of the Learning Sciences, Bloomington, USA*.

Blikstein, P. & Wilensky, U. (2007). *Bifocal modeling: a framework for combining computer modeling, robotics and real-world sensing.* Paper presented at the annual meeting of the American Educational Research Association (AERA 2007), Chicago, USA.

Blikstein, P., & Wilensky, U. (2008). *Implementing multi-agent modeling in the classroom: Lessons from empirical studies in undergraduate engineering education*. In M. Jacobson (Chair) & R. Noss (Discussant), Complex systems & learning: Empirical research, issues & "seeing" scientific knowledge with new eyes." In G. Kanselaar, J. van Merri'nboer, P. Kirschner, & T. de Jong (Eds.), *Proceedings of the International Conference of the Learning Sciences (ICLS)* (Vol. 3, pp. 266-267). Utrecht, The Netherlands

Blikstein, P. & Wilensky, U. (2009). An atom is known by the company it keeps: A constructionist learning environment for materials science using multi-agent modeling. *International Journal of Computers for Mathematical Learning*, *14*(2), 81–119.

Blikstein, P. (2009). An atom is known by the company it keeps: Content, representation and pedagogy within the epistemic revolution of the complexity sciences. Unpublished Doctoral Dissertation, Northwestern University, Evanston, IL.

Bransford, J. D., & Schwartz, D. L. (1999). Rethinking transfer: A simple proposal with multiple implications. *Review of Research in Education, 24*, 61–100.

Brown, G. S. (1961). Today's dilemma in engineering education. *IRE Transactions on Education, E4*(2), 48.

Burke, J. E. (1949). Some factors affecting the rate of grain growth in metals. *Transactions of the American Institute of Mining Engineers* (180), 73–91.

Centola, D., Wilensky, U., & McKenzie, E. (2000, June 14–17). A hands-on modeling approach to evolution: Learning about the evolution of cooperation and altruism through multi-agent modeling – The EACH Project. *Proceedings of the Fourth Annual International Conference of the Learning Sciences, Ann Arbor, MI*.

Colgate, J. E., McKenna, A., & Ankenman, B. (2004). IDEA: Implementing Blikstein & Wilensky design throughout the curriculum at northwestern. *International Journal of Engineering Education, 20*(3), 405–411.

Collier, N., & Sallach, D. (2001). *Repast*. Chicago: University of Chicago. Retrived from http://repast.sourceforge.net.

Committee on the Education and Utilization of the Engineering. (1985). *Engineering education and practice in the United States: Foundations of our techno-economic future*. Washington, DC: National Research Council.

diSessa, A. A. (2000). *Changing minds: Computers, learning, and literacy*. Cambridge: MIT Press.

Dym, C. L. (1999). Learning engineering: Design, languages, and experiences. *Journal of Engineering Education, 88*, 145–148.

Dym, C. L., Agogino, A. M., Eris, O., Frey, D. D., & Leifer, L. J. (2005). Engineering design thinking, teaching, and learning. *Journal of Engineering Education, 94*(1), 103–120.

Edwards, L. (1995). Microworlds as representations. *Proceedings of the 2nd International NATO Symposium on Advanced Technology and Education*.

Einstein, H. H. (2002). Engineering change at MIT. *Civil Engineering, 72*(10), 62–69.

Gobert, J., Horwitz, P., Tinker, B., Buckley, B., Wilensky, U., Levy, S., et al. (2004, July 31–August 2). *Modeling across the curriculum: Scaling up modeling using technology.* Paper presented at the Proceedings of the Twenty-fifth Annual Meeting of the Cognitive Science Society, Boston, MA.

Goldstone, R. L., & Wilensky, U. (2008). Promoting transfer through complex systems principles. *Journal of the Learning Sciences, 15*, 35–43.

Haghighi, K. (2005). Systematic and sustainable reform in engineering education. *Journal of Environmental Engineering-ASCE, 131*(4), 501–502.

Hatano, G., & Oura, Y. (2003). Commentary: Reconceptualizing school learning using insight from expertise research. *Educational Researcher, 32*(8), 26–29.

Horwitz, P., Gobert, J., Wilensky, U., & Dede, C. (2003). *MAC: A longitudinal study of modeling technology in science classrooms.* Paper presented at the National Educational Computing Conference (NECC), Seattle, WA

Hurst, K. D. (1995). *A new paradigm for engineering education.* Paper presented at the 25th Annual ASEE/IEEE Conference: Frontiers in Education '95, Atlanta, GA

Jerath, S. (1983). Engineering-education in perspective. *Mechanical Engineering, 105*(2), 92–93.

Katehi, L., Banks, K., Diefes-Dux, H., Follman, D., Gaunt, J., Haghighi, K., et al. (2004). *A New Framework for Academic Reform in Engineering Education*. Paper presented at the 2004 American Society for Engineering Education Conference, Salt Lake City, UT.

Krause, S. J., & Tasooji, A. (2007). *Diagnosing students' misconceptions on solubility and saturation for understanding of phase diagrams.* Paper presented at the American Society for Engineering Education Annual Conference, Portland, USA

Kulov, N. N., & Slin'ko, M. G. (2004). The state of the art in chemical engineering science and education. *Theoretical Foundations of Chemical Engineering, 38*(2), 105–111.

Lamley, R. A. (1996). A model for an engineering education. *Marine Technology and Sname News, 33*(2), 119–121.

Levy, S. T., Kim, H., & Wilensky, U. (2004, April 12–16). *Connected chemistry - A study of secondary students using agent-based models to learn chemistry.* Paper presented at the Annual meeting of the American Educational Research Association, San Diego, CA

Levy, S. T., Novak, M., & Wilensky, U. (2006). *Students' foraging through the complexities of the particulate world: Scaffolding for independent inquiry in the connected chemistry (MAC) curriculum.* Paper presented at the Annual Meeting of the American Educational Research Association, San Francisco, CA

Martin, F. (1996). Ideal and real systems: A study of notions of control in undergraduates who design robots. In Y. Kafai & M. Resnick (Eds.), *Constructionism in practice* (pp. 297–332). Mahwah, NJ: Lawrence Erlbaum Associates Inc.

MIT Center for Policy Alternatives. (1975). Future directions in engineering-education - system response to a changing world - report by center for policy alternatives for MIT School of Engineering. *Engineering Education, 65*(5), 382–382.

Minsky, M. (1986). The Society of Mind. New York: Simon & Schuster.

Newstetter, W. C., & McCracken, M. (2000). *Design learning as conceptual change: A framework for developing a science of design learning. Conference of the American Society of Engineering Education*. St. Louis, MO: ASEE.

Ono, N., Kimura, K., & Watanabe, T. (1999). Monte Carlo simulation of grain growth with the full spectra of grain orientation and grain boundary energy. Acta Materialia, *47*(3), 1007–1017.

Pagels, H. R. (1988). *The dreams of reason: The computer and the rise of the sciences of complexity*. New York: Simon & Schuster.

Panel on Undergraduate Engineering Education. (1986). *Engineering undergraduate education*. Washington, DC: National Research Council.

Papert, S. (1980). *Mindstorms: children, computers, and powerful ideas*. New York: Basic Books.

Papert, S. (1991). Situating constructionism. In S. Papert & I. Harel (Eds.), *Constructionism*. Cambridge, MA: MIT Press.

Parunak, H. V. D., Savit, R., & Riolo, R. L. (1998). Agent-based modeling vs equation-based modeling: A case study and users' guide. *Lecture Notes in Computer Science, 1534*, 10–25.

Porter, D. A., & Easterling, K. E. (1992). *Phase transformations in metals and alloys* (2nd ed.). London; New York: Chapman & Hall

Roco, M. C. (2002). Nanotechnology - A frontier for engineering education. *International Journal of Engineering Education, 18*(5), 488–497.

Russell, J. S., & Stouffer, W. B. (2005). Survey of the national civil engineering curriculum. *Journal of Professional Issues in Engineering Education and Practice, 131*(2), 118–128.

Sengupta, P. & Wilensky, U. (2008). *On the learnability of electricity as a complex system*. In M. Jacobson (Chair) & R. Noss (Discussant), Complex systems & learning: Empirical research, issues & "'seeing" scientific knowledge with new eyes. In G. Kanselaar, J. van Merri'nboer, P. Kirschner, & T. de Jong (Eds.), *Proceedings of the International Conference of the Learning Sciences (ICLS)* (Vol. 3, pp. 267-268). Utrecht, The Netherlands

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), 1–61.

Siegler, R. S., & Crowley, K. (1991). The microgenetic method: A direct means for studying cognitive development. *American Psychologist, 46*(6), 606–620.

Soloway, E., Guzdial, M., & Hay, K. E. (1994). Learner-centered design: the challenge for HCI in the 21st century. *Interactions, 1*(2), 36–48.

Srolovitz, D. J., Anderson, M. P., Grest, G. S., & Sahni, P. S. (1984). Computer simulation of grain growth - II. Grain size distribution, topology and local dynamics. *Acta Metallurgica, 32*, 793–801.

Stieff, M., & Wilensky, U. (2003). Connected chemistry: Incorporating interactive simulations into the chemistry classroom. *Journal of Science Education and Technology, 12*(3), 285–302.

Thornton, K., & Asta, M. (2005). Current status and outlook of computational materials science education in the US. *Modelling and Simulation in Materials Science and Engineering, 13*(2), R53.

Tisue, S., & Wilensky, U. (2004, May 16–21). NetLogo: A simple environment for modeling complexity. *Paper presented at the International Conference on Complex Systems (ICCS 2004), Boston, MA*

Tryggvason, G., & Apelian, D. (2006). Re-engineering engineering education for the challenges of the 21 st century. *JOM, 58*(10), 14–17.

Tyack, D. B., & Cuban, L. (1995). *Tinkering toward utopia: A century of public school reform*. Cambridge, MA: Harvard University Press.

Wilensky, U. (1993). *Connected mathematics: Building concrete relationships with mathematical knowledge.* Unpublished doctoral dissertation, MIT, Cambridge, MA

Wilensky, U. (1995). Paradox, programming and learning probability. *Journal of Mathematical Behavior, 14*(2), 231–280.

Wilensky, U. (1999a). GasLab-an extensible modeling toolkit for exploring micro-and-macro-views of gases. In N. Roberts, W. Feurzeig & B. Hunter (Eds.), *Computer modeling and simulation in science education*. Berlin: Springer Verlag.

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. (2006). *Complex systems and restructuration of scientific disciplines: Implications for learning, analysis of social systems, and educational policy.* In J. Kolodner (Chair), C. Bereiter (Discussant), & J.D. Bransford (Discussant), Complex systems, learning, and education: Conceptual principles, methodologies, and implications for educational research. Paper presented at the Annual Meeting of the American Educational Research Association, San Francisco, CA

Wilensky, U., Hazzard, E., & Longenecker, S. (2000, October 11–13). *A bale of turtles: A case study of a middle school science class studying complexity using StarLogoT.* Paper presented at the meeting of the Spencer Foundation, New York

Wilensky, U., & Papert, S. (in preparation). Restructurations: Reformulations of knowledge disciplines through new representational forms. Working Paper. Evanston, IL. Center for Connected Learning and Computer-Based Modeling, Northwestern University.

Wilensky, U., Papert, S., Sherin, B., diSessa, A., Kay, A., & Turkle, S. (2005). *Center for Learning and Computation-Based Knowledge (CLiCK). Proposal to the National Science Foundation Science of Learning Centers.*

Wilensky, U., & Rand, W. (2009). *An introduction to agent-based modeling: Modeling natural, social and engineered complex systems with NetLogo*. Cambridge, MA: MIT Press.

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

Wilensky, U., & Resnick, M. (1999). Thinking in levels: A dynamic systems approach to making sense of the world. *Journal of Science Education and Technology, 8*(1), 3–19.

Wilkerson-Jerde, M., & Wilensky, U. (2009). *Complementarity in equational and agent-based models: A pedagogical perspective.* Paper presented at the Annual Meeting of the American Research Education Association, San Diego, CA

Wilensky, U. (2006). Complex systems and reformulating scientific disciplines: Implications for learning, analysis of social systems, and educational policy. Paper presented at the American Educational Research Association, San Francisco, CA.

Wolfram, S. (2002). *A new kind of science*. Champaign, IL: Wolfram Media.

Yamins, D. (2005). *Towards a theory of "local to global" in distributed multi-agent systems (I).* Proceedings of the Fourth International Joint Conference on Autonomous Agents and Multiagent Systems, Utrecht, The Netherlands.

Software and Model References

Blikstein, P. & Wilensky, U. (2005b). NetLogo MaterialSim Grain Growth model. Evanston, IL: Center for Connected Learning and Computer-Based Modeling, Northwestern University. http://ccl.northwestern.edu/netlogo/models/MaterialSimGrainGrowth.

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

Wilensky, U., & Shargel, B. (2002). *Behavior Space [Computer Software]*. Evanston, IL: Center for Connected Learning and Computer Based Modeling, Northwestern University.

Blikstein, P., & Wilensky, U. (2004b). MaterialSim curriculum. Center for Connected Learning and Computer Based Modeling, Northwestern University, Evanston, IL. Retrieved from http://ccl.northwestern.edu/curriculum/materialsim.