The Application of System Dynamics Modeling in Elementary and Secondary School Curricula

Stephen Alessi

The University of Iowa

steve-alessi@uiowa.edu

Abstract

System Dynamics is a methodology for analysis, problem solving, and simulation development. This paper describes the System Dynamics methodology, its history, resources for learning more about it, several projects applying System Dynamics to education, and some of the important issues for educators who plan to apply it in their teaching. Recommendations for getting started are provided.

History and Introduction

System Dynamics is a methodology for analyzing complex systems and problems with the aid of computer simulation software. Its use in education is rapidly increasing as more and more educators discover that it is an excellent way to teach thinking and problem-solving in a systematic way with good feedback for learners.

System Dynamics was formulated by Jay Forrester in the 1960s at M.I.T. (Forrester, 1961, 1968, 1969, 1971). Forrester, a professor in M.I.T.'s Sloan School of Management, became interested in the complexity of business management and the forces that caused businesses to succeed or fail. He concluded that people are not good at dealing with complex systems in which many factors influence outcomes, such as the success of a business depending on employees, consumers, middlemen, the economy, and the weather (in agricultural businesses),to name just a few.

Forrester observed that people usually identify one or two influences and assumed that those account for the observed outcomes or a problem. As a result, people implement simple policies for solving a problem or reaching a goal, and quite often those polices have the opposite effect desired, the problem becomes worse, the business fails, and so on. A primary reason for this is policy resistance, namely, other people or influences in the system act to counteract the simple changes instituted and the intended outcome. There are many examples of this. For example, lowering the price of consumer goods does not necessarily increase consumer sales for a business, because competing businesses lower their own prices in response.

To help improve decision making and policy formation, Forrester created the System Dynamics methodology, an approach for analyzing complex systems to include all the relevant cause-effect relationships, and more important, time delays and feedback loops in those systems which account for most of their unexpected behavior. DYNAMO (Pugh, 1983), a mainframe computer program, was developed to facilitate creating simulation models of systems. DYNAMO permitted business managers to experiment, investigating potential solutions to problems and their likely outcomes. With System Dynamics modeling, Forrester demonstrated how simple problem solutions often had unintended and undesirable effects, and how problems could be better solved with more sophisticated levels of analysis.

Originally, Forrester applied System Dynamics to modeling and problem solving in industrial corporations (Forrester, 1961). Subsequently, he generalized the approach and applied it to social issues such as economics, crime and health (Forrester, 1969) and later to the physical and biological sciences such as ecology (Forrester, 1971).

For years, System Dynamics was the domain of university academics and researchers, requiring large mainframe computers to create and run complex DYNAMO models. But microcomputers became available in the late 1970s and in the early 1980s Micro-DYNAMO (Pugh-Roberts, 1982a,b) made System Dynamics modeling possible for everyone with an inexpensive Apple or IBM microcomputer. Shortly thereafter, *Computer Simulation: A System Dynamics Modeling Approach* was published (Roberts, et al., 1983), a textbook for secondary schools and colleges which taught the application of System Dynamics modeling to a wide variety of academic subjects including biology, psychology, physics, ecology, health science, economics, and mathematics. In fairly simple language it explained how successful problem solving in complex systems requires understanding the whole system, not just some small part of it, how such understanding could be attained through System Dynamics modeling, and how problem solving could be improved through its use.

In the years since, far more powerful computer programs for System Dynamics modeling have been created for both Windows and Macintosh computers, including PowerSim (PowerSim, 1999), STELLA (High Performance Systems, 2000b), ithink (High Performance Systems, 2000a), Extend (Imagine That, 2000), and Vensim (Ventana Systems, 1999). STELLA, which is available for both Windows and Macintosh, is especially suited for elementary and secondary school students. In edition, an even simpler program for System Dynamics modeling called Model-It (Soloway, et al., 1997) has been created at the University of Michigan for use by elementary and secondary school students.

More recent materials and textbooks for learning about System Dynamics include the following:

> *STELLA Introduction to Systems Thinking*, which is packaged with STELLA 6 (High Performance Systems, 2000b), the latest version, and is suitable for secondary or college level students in many subject areas.
>
> *Business Dynamics* (Sterman, 2000), a college and graduate-level textbook oriented towards business students.
>
> *Systems thinking and modeling: Understanding change and complexity* (Maani & Cavana, 2000), a general textbook intended for college students.
>
> *Dynamic modeling* (Hannon & Ruth, 1994), also a college level text appropriate in a variety of disciplines.
>
> *Principles of System Dynamics* (Bala, 1999), another domain general college textbook.
>
> *Modeling the environment: An introduction to System Dynamics modeling of environmental systems* (Ford, 1999), a college and graduate level text specific to the environmental sciences.

Most of these materials are oriented to college level students and higher. That means they may be good materials for educators to learn about System Dynamics modeling, but (with the exception of *STELLA Introduction to Systems Thinking),* might not be appropriate for elementary and secondary level students. However, a variety of materials are available on Web sites for both teacher training, and younger students' learning. They include the following Web sites:

> www.hi-ce.org/sciencelaboratory/modelit The Model-It project
>
> www.teleport.com/~sguthrie The CC-SUSTAIN project
>
> www.hps-inc.com High Performance Systems, Inc. (STELLA)
>
> www.powersim.com Powersim, Inc. (Powersim)
>
> www.imaginethatinc.com Imagine That, Inc. (Extend)
>
> www.vensim.com Ventana Systems, Inc. (Vensim)
>
> http://sysdyn.mit.edu/ M.I.T. System Dynamics in Education Project

Three substantial curriculum projects have focused on the use of System Dynamics modeling in elementary and secondary schools. They are the Model-It project at Michigan's Highly Interactive Computing Laboratory (Soloway et al., 1997), the CC-SYSTAIN project funded by the National Science Foundation (Zaraza & Fisher, 1997; Zaraza, Joy, & Guthrie, 1998), and the STACIN project at Educational Testing Service (Mandinach & Cline, 1994, 1996). The first two can be investigated through their Web sites identified above. The last may be studied by reading the book by Mandinach & Cline (1994) or contacting E.T.S.

There is also a growing research and application literature for System Dynamics applied at many educational levels, including several of the chapters in Feurzeig & Roberts (1999), and the many papers at the annual conference of the System Dynamics Society (www.albany.edu/cpr/sds/) which are published in its proceedings (e.g., Davidsen, Ford, & Mashayekhi, 2000). The 2001 conference of the System Dynamics Society will be in Atlanta, Georgia, U.S.A.

To summarize, System Dynamics is a well formulated methodology for analyzing the components of a system including cause-effect relationships and their underlying mathematics and logic, time delays, and feedback loops. It began in the business and industry world, but is now affecting education and many other disciplines. More and more people are beginning to appreciate the ability of the System Dynamics methodology to bring order to complex systems and to help people learn and understand such systems.

Having introduced the history and basic philosophy of System Dynamics, I will now do four things. First, I will describe three interesting curriculum projects which implement System Dynamics in elementary and secondary school education. Second, I will describe my own approach which attempts to integrate both using simulations and creating System-Dynamics-Based simulations for learning. Third, I will discuss some important issues to be considered if you pursue the use of System Dynamics for education. Fourth and last, I will conclude with recommendations for getting started with System Dynamics if you wish to do so.

System-Dynamics-Based Curriculum Projects

Although there are others, I will describe three projects in the United states which are representative of the various ways the System Dynamics approach is being applied in pre-college education.

STACIN. The STACIN project (which stands for Systems Thinking and Curriculum Innovation Network) grew out of the Apple Classroom of Tomorrow (ACOT) project of the 1980s. The ACOT project provided schools with significant computer resources, enabling them to implement curriculum innovations which depended on intensive computer use. Combining that with funding from the U.S. Department of Education and from Educational Testing Service, it began as the STACI project with one high school in the state of Vermont, and became STACIN as six schools were added in the San Francisco area (four secondary schools and two elementary schools) and one high school was added in Arizona. As of their latest reports, approximately forty teachers have been trained to implement System Dynamics thinking in their curricula, ranging from about grades five to twelve. In the elementary schools (grades five through eight in this project) the curricula areas have been science, math, and social studies. In the high schools (grades nine through twelve), the humanities have been included.

Mandinach & Cline (1994) describes the early and middle years of the project in some detail. The driving philosophy was to implement curriculum of a constructivist nature, in which learners and teachers worked collaboratively to create elaborated understanding of systems and processes, and to solve problems. In the beginning the project was somewhat less constructivist, attempting to teach various principles and systems to students. But, in part due to their early experiences and in part due to the changing educational milieu in the United States, they began to implement a more learner-centered approach in which greater emphasis was on the students creating their own models. Their guiding philosophy was that knowledge should not and cannot be "poured" into students' heads. Rather, students and teachers must create, expand, and refine their knowledge together.

STACIN uses the STELLA software. At the start of the project STELLA operated only on Apple Macintosh computers, but it has since been ported to work identically on the Windows operating system. STELLA allows the learner to create a System Dynamics model by dynamically building a diagram, such as the one illustrated in Figure 1. This is done by dragging and dropping pre-constructed objects.

The four main objects, which are the main objects of all System Dynamics software packages (through their names and appearance may vary slightly), are stocks (such as the rectangle labeled *heat in coffee*), flows (the double line with a controller, labeled *heat loss*), converters (the circles labeled *temperature of coffee*, *volume of coffee*, *room air temperature*, and *temperature difference*), and cause-effect arrows, such as the one pointing from *temperature difference* to *heat loss*. In general, a stock represents a quantity of something (such as the quantity of heat in a cup of coffee). A flow represents the rate of change (the rate of increase and/or decrease) of a stock. A converter represents any other constant or variable which affects (or is affected by) other components of the system.

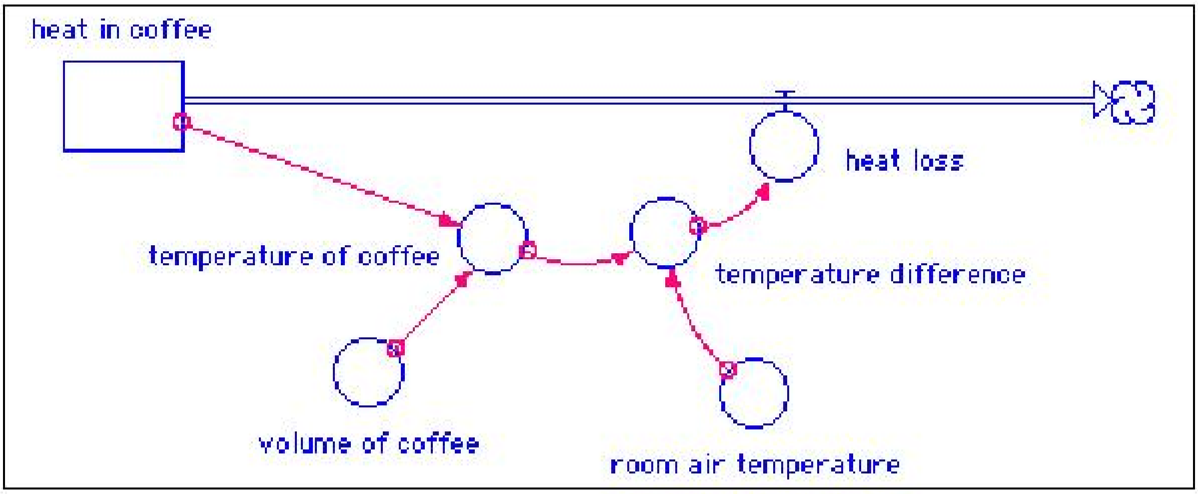

Figure 1. Flow diagram of coffee cooling.

There are a variety of secondary objects in STELLA, mostly for the purposes of user input, output, and control. There are numeric graphs, tables, and fields to show how numbers change as the parts of the system interact. There are sliders, buttons, and switches to allow the learner to control input numbers and conditions.

The process of creating a model is an iterative one. You create an initial diagram (generally based on research and a hypothesis about how the system operates). By double-clicking on stocks, flows, and converters, you open up associated windows in which you can enter initial values, variables, and formulas. These may be constrained by cause effect lines. That is, the formula in the *temperature of coffee* converter must include mention of both *heat in coffee* and *volume of coffee* because both of those point to it. Similarly, the flow labeled *heat loss* must include mention of *temperature difference* because there is a cause-effect arrow connecting them.

After creating a diagram and filling in formulas, you may run the model and see its behavior. For example, Figure 2 illustrates a graph showing the manner in which hot coffee will cool according to the model in Figure 1. The coffee will cool quickly at first, but then cool more slowly as it approaches the temperature of its surroundings. Based on the model's behavior (whether it works at all, whether the behavior resembles the real world) you may need to do more research, alter your hypotheses, and modify the diagram. Modification may include adding or deleting stocks, flows and converters; adding, deleting, or re-routing cause-effect arrows; and changing the formulas within each stock, flow, or converter.

The iterative process of improving the model continues, and does not necessarily have a distinct stopping point. Models and diagrams are never really perfect. The real learning benefit is in the process of creating and refining the model, not the end product.

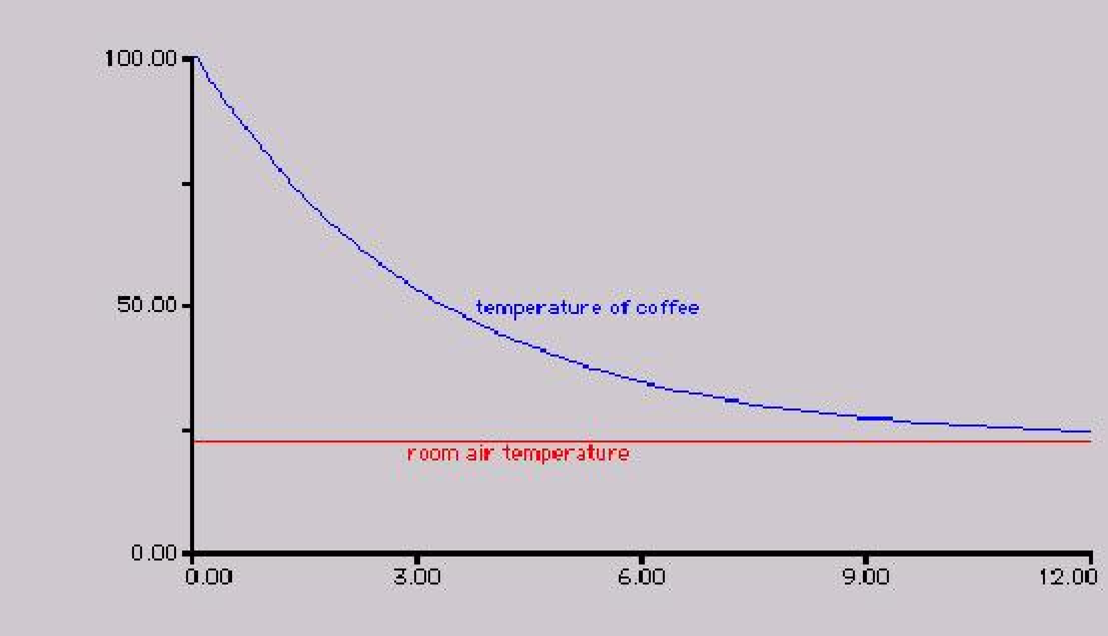

Figure 2. Graph of coffee cooling.

Mandinach & Cline (1994) describes how the above process of using STELLA was applied in various subject areas, the initial problems in the project and how they solved them, and their teaching methodology. The main thing worth noting about their teaching methodology is that they advocate four stages. In the first stage, *parameter manipulation*, students manipulate the numbers in existing simulation models created by other people. In the second stage, *structural manipulation*, students modify the actual models, that is, they add, delete, change, or reorder stocks, flows, converters, cause-effect arrows, and the numbers within them. In the third stage, *constrained modeling*, learners construct their own simple models, like the coffee cooling model illustrated in Figure 1. In the fourth and last stage, *epitome modeling*, learners engage in the entire sequence of creating an initial model, evaluating it, and improving it through revisions. This four-phase process is worth noting because there is disagreement among people who practice System Dynamics concerning the best way to facilitate students learning it. Some people believe that learners should be started immediately in the third phase, creating models, rather than analyzing models created by other people.

CC-SUSTAIN.

The CC-SUSTAIN project has goals and characteristics similar to the STACIN project, but also has notable differences. It began as the CC-STADUS project (Cross-Curricular System Thinking and Dynamics Using STELLA) in 1993 with three years of funding from the United States National Science Foundation (NSF). They initially trained about 150 high school teachers to incorporate System Dynamics modeling with STELLA in their curricula. Teachers were trained during summer workshops. Subsequently, teachers worked with their students (all at the high school level) to create models in many areas, but again, primarily science, math, and social studies. Examples in social studies include models of land use and housing availability, population change, and slavery. Examples in the physical sciences include the orbits of planets and motion of falling objects. Examples in the biological and health sciences include alcohol consumption and its effects on the body, ecology, the spread of infectious diseases, and drug addition. Examples in mathematics include simple algebra (compound interest in a bank account) and more complex calculus (the trajectories of flying projectiles). One example from the humanities is an analysis of the growth of savagery as reflected in William Golding's novel *Lord of the Flies*.

After the initial three years of the CC-STADUS project, an additional three year grant was awarded by the National Science Foundation to emphasize continued teacher training and dissemination of the project. Thus, CC-SUSTAIN (Cross-Curricular Systems Using Stella: Training and INservice) became the new and current project name. Over 240 teachers have now been trained to use System Dynamics thinking in their teaching and curricula.

The notable differences between STACIN and CC-SUSTAIN should be pointed out. STACIN is more a research project. It has included a relatively small number of teachers and schools and its primary goals have been investigating and evaluating the educational effects of implementing System-Dynamics-Based learning environments. It has been directed by a few project leaders with a fairly unified approach to structuring the learning environments (the four stages described earlier). It has included both elementary school and high school students.

In contrast, CC-SUSTAIN is primarily an implementation (rather than a research) project. It has trained many more teachers in more schools. Because more people have been involved, it has had more diversity of educational philosophy. For example, some teachers in the CC-SUSTAIN project (Fisher, 1998) subscribe to the point of view that learners should start modeling immediately (the third phase of the STACIN approach) rather than analyzing or modifying models created by other people. Although there has been more diversity of approaches in CC-SUSTAIN, its philosophy generally includes the following: Encourage learners to start modeling immediately. Begin with very simple models and build upon them. Involve only teachers eager to embrace System Dynamics thinking. (That is, do not try to evangelize and get every teacher in a school to join the project.) Allow System Dynamics to change your approach to teaching, rather than trying to fit it into your current method of teaching. Use System Dynamics to open up avenues for student research and independent learning. Lastly (though there is considerable difference of opinion on this point), emphasize System Dynamics as a methodology of solving problems in systems, rather than as an approach for modeling a system in general. I will return to this last consideration when I discuss my own use of System Dynamics in education.

For teachers considering the use of System Dynamics, the STACIN project, being research and evaluation, should be considered as an indicator of *whether* it is worth integrating System Dynamics in elementary and high school teaching. In contrast, the CC-SUSTAIN project, having implementation as its main goal, should be considered for *how* to go about implementation, if you decide it is worth doing.

As of today, the jury is still out on the former question, whether or not System Dynamics in elementary and high school curricula is worthwhile. Although Mandinach & Cline portray the project positively, primarily with descriptive anecdotes of individual teacher success stories, they have not yet reported a final evaluation or judgement of the approaches value. However, if you feel inclined to pursue System Dynamics as these two projects portray it, the materials and suggestions of the CC-SUSTAIN project (available on their Web site and in their summer training institutes) are particularly worthwhile.

Model-It.

Model-It is a software component of the ScienceWare project at the University of Michigan's Highly Interactive Computing (Hi-C) research laboratory. Unlike the previous two projects, which were multidisciplinary, ScienceWare and Model-It are designed specifically for learning science, primarily at the secondary school level. The underlying philosophy and assumptions of the ScienceWare project are that modeling is central to the doing and learning of science, that modeling includes several important components (collecting data, visualizing data, creating models to represent and explain data, and reporting the data), and that student motivation for modeling cannot be assumed.

Several tools were created for the ScienceWare project including RiverBank for data collection, Viz-It for data visualization, Model-It for model creation and refinement, PlanIt-Out for scientific research planning, and Web-It for reporting and publishing results.

Like the STACIN and CC-SUSTAIN projects, ScienceWare and Model-It are based on constructivist foundations. As such, authentic problems (the type that professional scientists deal with) contextualized in the students' real world are pursued. Important and sometimes complex problems are tackled, though they must be ones that are feasible for novices to be successful with and must be interesting to them. Usually the problems have a social focus, such as the impact of pollution on the local environment or the effect of drug addition on their friends and neighborhoods. There is an emphasis on students constructing knowledge and being able to demonstrate their understanding through application and explanation of the knowledge. The activities emphasize students planning research strategies and tactics and working collaboratively within a group of learners with varying expertise in different skills or aspects of the content. Lastly, technological tools to facilitate cognition (such as RiverBank, Viz-It, Model-It, PlanIt-Out, and Web-It) are provided to and used by students throughout their projects. Those programs are designed to facilitate multiple methods of representing information, including textual, graphical, pictorial, and auditory modes.

The designers of the ScienceWare project believed that System Dynamics modeling was an important component of learning and doing science. But they believed that software such as STELLA and Powersim were too complex and difficult for many secondary-level students to use. They liked the ease of use of pre-built simulations like SimEarth (Electronic Arts, 1998), but did not like that students could not see and manipulate the underlying models of such programs. In contrast, they liked the modeling process of programs like STELLA but found them too difficult for many of their students to use. They created Model-It to combine the best of both approaches, easy to use but making the models accessible to students.

Model-It was designed to be easier in a number of ways. It was created explicitly for science. Being less general than programs such as STELLA, it could have fewer components and therefore be simpler. It was not only visual like STELLA, but *pictorial* in its orientation. The visuals of STELLA are mostly abstract while those created for Model-It were more realistic in nature, thus being more concrete for learners. Additionally, the designers of Model-It built it with educational *scaffolding*, that is, they provided within the program instructional supports to guide students through their learning activities. The scaffolding is of three sorts, providing connections to the learners' prior knowledge and experience, providing information with multiple representations, and making clear the connection of actions and their effects. Examples of the first (prior knowledge connections) are providing learners with pre-defined objects (like a construction set), providing relevant and realistic photographs and graphic images, and at the start of a project providing clear verbal and qualitative explanations of relationships. Examples of the second (using multiple representations of data) include providing data in textual, numeric, graphic, and other visual forms; beginning with qualitative relationships and moving gradually to quantitative ones, and beginning with very concrete representations and moving gradually to more abstract ones. Examples of the third (making clear the connection between actions and effects) include allowing students to manipulate variables and other aspects of models directly while the model is being run by the computer, and providing immediate visual feedback (again, using multiple representations such as both numbers and graphs) showing the results of their variable manipulations.

The designers have reported on students using Model-It to do research on a variety of science topics including ecology (effects of pollution), health (effects of drug addiction), and weather. Their evaluation efforts have been mostly descriptive and qualitative, but they claim that modeling has helped students refine and better articulate their understanding of science, and that learners find modeling interesting and are even motivated to continue extending and enlarging their models to encompass more data and situations. They attribute much of the success students have with Model-It to its instructional supports in the form of scaffolding.

Summary.

These three projects represent good examples of how System Dynamics may be used in school curricula. The STACIN project provides a fairly concise model (the four phases) for using System Dynamics and associated software with students. It provides some, albeit limited, evidence that System Dynamics does improve learning and attitudes. The CC-SUSTAIN project provides extensive materials for teacher and student use. (In contrast, the STACIN project has disseminated few such materials.) It also continues to provide summer institutes for teachers wishing to learn the application of System Dynamics in their curricula. It does not suggest so concise a model as does STACIN. Rather, reflecting the large number of teachers involved, it illustrates many different approaches to applying System Dynamics. The Model-It part of the ScienceWare project suggests a more learner friendly approach based upon a well-formulated theory, as reflected in the extensive use of scaffolding, collaborative learning, and other features of their philosophy. However, it has been limited to science education and the materials may be more difficult to apply in math, social studies, or humanities curricula. Taken together, these projects provide a number of useful models and ideas for any teacher wishing to experiment with System Dynamics in the classroom.

Integrating Simulation Use with Model Building

The projects just discussed are based on the notion that students learn better when building their own models of knowledge than by studying the models built by teachers and other people. This is directly stated in writings of the CC-SUSTAIN project and a little more obliquely suggested in those of the STACIN and Model-It projects. This is not a position that I completely share. As I have argued elsewhere (Alessi, 2000a), some content (for example, much procedural knowledge) may be better learned by using models and simulations constructed by experts, while other content (usually conceptual and declarative knowledge) may be better learned by building models in a System Dynamics environment. Furthermore, characteristics of learners, such as their age, motivation, and previous experience with the subject matter, may differentially suggest building versus using simulations. There are other considerations, for example, the importance of performance, cost efficiency, and time constraints, which also suggest one approach over the other. As a consequence of these many considerations, I maintain that a *combination* of using simulations and building simulations is appropriate for most learners and educational environments.

In my graduate teaching (Alessi, 2000b) and in research (Quinn & Alessi, 1994) I have observed students learn a significant amount in complex subject matter both using simulations created by other people and by creating their own simulations. The simulations designed by other people may be created using System Dynamics theory and software, and therein lies another purpose for System Dynamics programs such as STELLA and Powersim. Not only can they be used as learning tools themselves, but they can be used as instructional design tools to create educational simulations for use by students. Whereas none of the projects discussed above (or that I am in general aware of) has attempted to apply System Dynamics thinking and software to very young students (that is, below grade 5) simulations built using STELLA can be designed for those levels. In particular, System Dynamics software is useful for creating educational simulation of the Scientific Discovery Learning variety (de Jong & van Joolingen, 1998). These are simulations which embody a model of a scientific theory or phenomenon, and allow the student to engage in simulated research by formulating hypotheses, doing experiments by systematically manipulating variables, collecting and analyzing data, and making interpretations. Although scientific discovery learning may use simulations designed with System Dynamics software, it is based on a very different educational philosophy. Scientific discovery learning confronts the student with "black box" simulations, in which the underlying model is hidden and the student must discover it through systematic research. In contrast, most use of System Dynamics for learning utilizes a "glass box" approach in which the model is completely visible at all times (Breuer, 2000; Wenger, 1987).

A very useful feature of STELLA and Powersim is that they make complex mathematical formulas available to the user. You can create a simulation by dragging and dropping icons and entering fairly simple algebra, after which these powerful programs create the complex differential equations necessary to actually run the model as a simulation. Furthermore, a designer or teacher can copy the equations from these packages into other computer programs designed for lesson design, such as Authorware (Macromedia, 1999) or ToolBook (Asymetrix Learning Systems, 1997). Figure 3 illustrates the equations generated by STELLA for the coffee cooling model illustrated earlier. These are simple text strings and are easily copied to almost any other computer software package.

```
{ initializations }
heat_in_coffee = 1000
volume_of_coffee = 10
temperature_of_coffee = heat_in_coffee / volume_of_coffee
room_air_temperature = 22
temperature_difference = temperature_of_coffee - room_air_temperature
heat_loss = temperature_difference * 3

{ run-time calculations }
heat_in_coffee(t) = heat_in_coffee(t - dt) + (- heat_loss) * dt
temperature_of_coffee = heat_in_coffee / volume_of_coffee
temperature_difference = temperature_of_coffee - room_air_temperature
heat_loss = temperature_difference * 3
```

Figure 3. Equations from the coffee cooling model.

Many educators who have been willing to tackle the creation of simple multimedia lessons, for example, tutorials, drills, and hypermedia, have been unwilling to attempt simulation design because of the added complexity of programming the underlying mathematical model of the phenomenon to be simulated. System Dynamics software makes creation of the mathematical model much easier. I suggest that for the lowest grade levels (grades 1 through 6) teachers and courseware designers should use System Dynamics software to create learning simulations appropriate for such young learners. These would be simulations for classroom demonstrations and for scientific discovery learning. For grades 5 and up (obviously there will be some grade overlap for these different approaches, and not a strict line of division), a combination of students *using* simulations built with the assistance of System Dynamics software, and actually *building* System Dynamics models themselves is possible and advantageous. We can go beyond the approaches used in projects such as STACIN, CC-SUSTAIN, and Model-It, to not only have students do modeling, but learn from complete and more sophisticated models built by experts. The former (building models) facilitates important thinking and problem-solving skills and gives learners a good understanding of basic concepts, while the latter (using pre-built models) makes a much wider range of complex phenomenon and topics available to them.

Issues in System-Dynamics-Based Learning Environments

I don’t want to create the impression that using the System Dynamics approach is easy. There are difficulties and issues to be aware of, and I discuss those next.

Difficulty of teaching System Dynamics.

Everyone who teaches System Dynamics modeling has reported how difficult it is, even though the benefits are great. There are errors all students make and difficulties they all encounter. Students tend to confuse stocks (levels) with flows (rates of change). They try to incorporate the formulas of previous science and math classes (which they often do not fully understand) instead of doing true system analysis. When models do not work correctly, they include *fudge factors*. Fudge factors are formulas, constants, or logical conditions designed to artificially fix the problem, not to realistically model the system. Students fail to test their models well, so the models tend to work only for common conditions, rather than the wide range of real-life conditions. Students confuse flows with cause-effect relationships. They create models that are unnecessarily complex and abstract, rather than having a close correspondence to reality. They try to copy and adapt models from instructors or textbooks, instead of thinking through the phenomenon and generating their own models from scratch. They engage in trial-and-error modifications in the vague hope that the results will come out right. They create initial models with too many components, when they should start with a very simple model and slowly build it up. They ignore the units of variables and as a result combine variables that have different units, such as feet per second and kilometers per hour. The main error made by teachers is thinking that students can create their first sophisticated models in a few weeks, when doing so (and overcoming all the above problems) may take several months. Patience, with yourself and with your students, is essential. Learning System Dynamics is slow in the beginning and it takes some time before there is visible payoff.

Whether to model problems or systems.

Since the early days of System Dynamics, a basic tenet has been that you must model a specific *problem*, not a general system. Earlier I illustrated a model for cooling coffee. A System Dynamics purist would probably say that I should not give students a task like "model the cooling of a hot cup of coffee," because it is too general and open-ended. Instead, give a problem like the following (an example adapted from the CC-SUSTAIN training materials): "You have just poured yourself a cup of hot coffee but because of an interruption you will not drink it for about 10 minutes. You also want cream in it, but want the coffee to be as hot as possible when you drink it. Should you put the cream in now or in ten minutes?" The more specific nature of a problem, it is claimed, constrains the modeling task to just those components (stocks, flows, converters, and cause-effect relationships) necessary to solve the problem and answer the question.

This dictum, model problems rather than general systems, works well when using System Dynamics in business management and most other professional areas. But it is a little more difficult when using System Dynamics as a general educational tool. It is especially difficult to adhere to when using System Dynamics software as an instructional design tool for building educational simulations for demonstrations or scientific discovery learning. There is considerable debate among educators, especially at the elementary and secondary levels, as to whether this principle should always be followed.

What software to use.

Just as everyone has a strong opinion about what computer manufacturer and what operating system is best, everyone has an opinion about what System Dynamics software is best: STELLA, Powersim, ithink, Extend, Model-It, or Versim (which are the ones I am familiar with). This discussion takes on an almost religious fervor, and there is often no sense arguing with people about it. You must choose a package intelligently, taking into consideration the following factors.

Your computer operating system

The grade level of your students

The subject areas to which modeling will be applied

Local availability of the software

Local availability of expertise and support for the software

Whether you will be putting models on the Web

Whether you will be transferring model equations to other software

Your budget

Amount and quality of documentation you require

The complexity of models to be created and the accuracy required

What learning materials to use.

Earlier I listed a number of resources (textbooks and Web sites) available for learning about System Dynamics. A common complaint is, "There are no textbooks that address the subject area that I teach." I encounter this not only when teaching System Dynamics, but also when teaching the design and development of educational multimedia, and more recently the design and development of educational Web sites. Math teachers do not want to read books about creating Web sites for foreign language instruction. Language teachers do not want to look at examples of multimedia instruction in the sciences. My response is twofold. First, many of the fundamental methods (of Web site design, of multimedia education design, or of System Dynamics modeling) are the same whether they are being applied to languages, science, math, or anything else. Secondly, to the extent that there *are* differences, we are more likely to *expand our creativity* when we look at applications in subject areas most unfamiliar to us. The math teacher studying a math multimedia lesson is covering already familiar ground. But the math teacher studying an art multimedia lesson is likely to encounter ideas and methods entirely unexpected, and eventually will be inspired with creative ways to apply those methods to math instruction. For that reason, you should consider all of the materials listed earlier for learning System Dynamics. The more difficult issue is what materials to use (or modify for use) with your younger students. Most of the textbooks have been designed for college and university level students, and are appropriate for your own learning as a professional educator. The exceptions are, *STELLA Introduction to Systems Thinking* (High Performance Systems, 2000b), and *Computer Simulation: A System Dynamics Modeling Approach* (Roberts, et al., 1983), which are designed for more novice learners. However, the former assumes you will be using STELLA and the latter assumes you are using Micro-DYNAMO, which is somewhat outdated. Several of the Web sites listed, especially those for CC-SUSTAIN and the M.I.T. System Dynamics in Education project, include materials useful with younger learners.

What subject areas to consider.

There is a tendency for System Dynamics enthusiasts to claim and believe it is useful for just about any academic area. But in fact, there have been very few demonstrations of its utility in the humanities and the arts. Its application has been primarily to (in approximate order of frequency) business and economics, ecology, manufacturing, public policy development, biological sciences, health sciences, engineering, social sciences, chemistry, mathematics, and physics.

System Dynamics is a quantitative methodology, and as such, has more obvious application to content areas which are amenable to mathematical models. That is partially reflected in the order of the topic list in the preceding paragraph. However, we should be careful not to equate the original System Dynamics methodology with the current System Dynamics software packages. STELLA, for example, has expanded beyond the mathematical methods and functions of the original System Dynamics and now includes many logical functions, statistical and probability functions, and queuing functions. The inclusion of logical (If – Then) functions permits STELLA (and some other System Dynamics packages) to be used for more qualitative modeling. For example, logical and probability functions (including those functions which select random numbers from flat, normal, and Poisson distributions) can be combined for creating psychological and sociological models, which are less amenable to the continuous simulation methods based on calculus.

So on the one hand, the more quantitative a subject area the more examples and expertise you will find in the System Dynamics literature and community. On the other hand, System Dynamics software includes features useful for more qualitative domains. Being a teacher of a more qualitative area does not mean you should discount the utility of System Dynamics for you, but you should be aware of its historically limited application in those areas and the paucity of help and examples you will find.

Summary Recommendations

Enthusiasm required. My first recommendation is borrowed from the CC-SUSTAIN project and their admonition not to attempt foisting the approach on teachers who are not interested in it. Although I believe the System Dynamics approach can be very beneficial, it will only be so when used by teachers who are enthusiastic about it. You should use it only if it is right for you. Furthermore, you should only encourage or train others to use it if it is right for them. This seems obvious enough, yet it is a rule often broken by enthusiastic newcomers to the approach.

Include assessment. Second, we lack strong evidence that the System Dynamics approach is beneficial for elementary and secondary school students. Anecdotal evidence is good, which gives reason to consider using the approach. But if you choose to try System Dynamics you are advised to include your own assessment of outcomes, including student learning, student attitudes, and teacher (including your own) attitudes. Use the results of your self-assessment to make changes and to decide whether it is worth your continued effort.

Learn System Dynamics yourself before teaching it to others. Although a common theme of today's constructivist thinkers is that teachers and students explore and learn together, I do not recommend this for System Dynamics. It is true that you will continue to learn it as you teach it, but don't begin without a reasonable level of expertise. You must be able to forestall common student errors (discussed below) and provide reasonable assistance to your students. System Dynamics and its associated software is sophisticated, and requires considerable real use before you will be able to teach others.

Use STELLA for K-12 learning environments. I have said before that everyone has a different opinion about what System Dynamics software is best. My opinion is that it depends on your audience and purposes. In business environments Powersim or Vensim might be preferable. But for younger learners I recommend STELLA. It has several advantages for younger learners. The software has a very friendly user interface. The documentation contains sections appropriate for younger students. There is more experience using STELLA with school age students than most other software. It is general enough to be used in a variety of subject areas. It can be used on either Windows or Macintosh computers and the most recent version includes the capability for running models on the World Wide Web. Lastly, the best instructional materials for school age students are designed with STELLA in mind.

Begin your own learning with *STELLA Introduction to Systems Thinking*, assuming you choose to use STELLA. Following from the previous recommendation, this reference (available from the company that designed and markets STELLA) is an excellent place to begin your own learning and exploration of System Dynamics. It is easy to read, contains examples from a variety of academic disciplines, and is closely tied to the terminology and functionality of the STELLA system.

What instructional materials to use with your students. This, unfortunately, is where a single, clear recommendation is impossible. For pre-college students there is a paucity of good materials. For the youngest students (for example, grades 1 through 5), you must create your own instructional materials or adapt those from the CC-SUSTAIN project. For slightly older students (grades 5 through 9) materials from CC-SUSTAIN will be useful, or *Computer Simulation: A System Dynamics Modeling Approach* (Roberts, et al., 1983) can be used with modification and teacher support (because it is written for use with Micro-DYNAMO, an outdated system). For older students (grades 9 through 12) you can use the Roberts textbook, the CC-SUSTAIN materials, or *STELLA Introduction to Systems Thinking.* For college students there are several more options, including Sterman (2000), Ford (1999), Hannon & Ruth (1994), or Maani & Cavana (2000), depending on the academic disciplines you are teaching.

Use the Mandinach & Cline learning sequence. Although this is a somewhat controversial issue, I recommend the four phases suggested in Mandinach & Cline (1994) which begin with students running pre-built models, progresses to students modifying pre-built models, next creating their own simple models, and finally engaging in the entire process of model design, evaluation, and revision. Although some teachers maintain that students should begin immediately building their own models, research is needed to determine which approach is better under which circumstances.

Teach System Dynamics for your students to learn and practice problem solving. I earlier expressed some disagreement with the usual System Dynamics dictum that you should model problems, not general systems. Nevertheless, when teaching students to use System Dynamics, this is generally a good principle It will help keep models from growing too large and complex, will provide a clearer goal for students to work towards, and will help structure the learning activities. Remember that I say this is *generally* a good rule. Do not be afraid to break it.

Have your students work in collaborative groups. Developing System Dynamics models is an activity that benefits greatly from students working in collaborative groups (Singhanayok & Hooper, 1998). An individual student will often get stuck and not know how to begin. An individual student, once started on a model, may be unable to judge it critically and make changes. Small groups come up with more ideas, are able to engage in better self-criticism, and bring a greater variety of talents to the activity. There are potential difficulties in collaborative group work. For example, some students may do most of the work while others watch. But these can be avoided by proper group composition, instruction on project management, and instructor supervision. The problems are outweighed by the benefits.

Begin with easily modeled and conceptualized domains. It is critical that students' first modeling activities not be overwhelming. It might be necessary for the first activity to be outside of the intended subject area. For example, even if your subject area is economics, the simple coffee cooling problem (or hot chocolate for children) is a useful first activity because it has very few components and yet contains some good lessons about System Dynamics. Another favorite activity for beginners is in the field of ecology -- the predator-prey relationship among animal species such as wolves and rabbits. In contrast to the six components of the coffee-cooling model, a basic predator-prey model has about 12 components. The former exhibits what we call *goal seeking* behavior (the coffee slowly approaches room air temperature) while the latter exhibits *oscillatory* behavior, with the numbers of rabbits and wolves rising and falling periodically. It is valuable to start with simple models which demonstrate particular System Dynamics principles and classic types of behaviors before attempting to model a complex phenomenon relevant to the subject matter being studied.

Forestall common errors. As discussed earlier, there are a number of typical errors students make when they are first learning System Dynamics. Teachers can do a lot to prevent bad habits from forming. Encourage students to begin with very simple models, evaluate them, and slowly increase their complexity. Encourage students to create models which have a close correspondence to real quantities, processes, and objects, rather than trying to use classical or abstract formulas and laws. Teach students to avoid using fudge factors. They should instead analyze and improve the whole system. Continually remind students to pay attention to units and maintain unit consistency. Help students to properly identify what are stocks versus what are flows (a distinction which is not always easy). Lastly, encourage thorough model testing, not only for common conditions but for uncommon (though realistic) conditions as well.

Be patient. Most importantly, do not expect yourself or your students to learn System Dynamics quickly. It requires a new way of thinking, so it takes time and practice. System Dynamics is not a methodology worth studying for its own sake, but for its potential to improve learning in other subject areas. It must be integrated into other subject area curricula rather than being its own topic. If students are to benefit from it, they must study and use it over an extended period of time in real academic disciplines and with a variety of problems and phenomena.

Conclusion

The System Dynamics method began in the 1960s in the field of business administration. It has since grown to include practitioners in many fields including the physical and social sciences, mathematics, law, medicine, and education. As an educational approach, a growing number of teachers have embraced System Dynamics as a way to revitalize their classrooms. Current software makes System Dynamics modeling accessible to students as low as grade 5. System Dynamics software is also useful to teachers and instructional designers who are creating professional simulation lessons. Teachers looking for new ways to enhance students' understanding of complex systems, problem solving skills, and formal thinking skills, should investigate System Dynamics as a tool that can help accomplish many of those goals. Integrating the System Dynamics approach into your teaching can be challenging and time consuming, but many teachers who have done so have become almost evangelical supporters. This attests to the methodology's ability to motivate teachers and students alike. Those interested in learning more about System Dynamics thinking and modeling are encouraged to investigate the books and Web sites previously cited, or to seek out a course. The Web site of the System Dynamics Society maintains a list of institutions [http://www.albany.edu/cpr/sds/sdcourses/] in about thirty different countries where you can learn more about it.

References

- Alessi, S. M. (2000a). Building versus using simulations. In J. M. Spector & T. M. Anderson (Eds.), *Integrated & holistic perspectives on learning, instruction & technology: Improving understanding in complex domains* (pp. 175-196). Dordrecht, The Netherlands: Kluwer.
- Alessi, S. M. (2000b). Designing educational support in system-dynamics-based interactive learning environments. *Simulation & Gaming, 31*(2), 178-196.
- Asymetrix Learning Systems. (1997). *ToolBook II Instructor*. [Computer software]. Bellevue, WA: Asymetrix Learning Systems.
- Bala, B. K. (1999). *Principles of System Dynamics*. Udaipur, India: Agrotech Publishing Academy.
- Breuer, K. (2000). *Readability of glass-box models as part of learning environments in vocational and economics education*. Paper presented at the 18th International System Dynamics Conference, Bergen, Norway, August 6-10.
- Davidsen, P. I., Ford, D. N., & Mashayekhi, A. N. (Eds.). (2000). *Proceedings of the 18th International Conference of the System Dynamics Society: Sustainability in the Third Millennium*. Bergen, Norway: The University of Bergen.
- de Jong, T., & van Joolingen, W. R. (1998). Scientific discovery learning with computer simulations of conceptual domains. *Review of Educational Research*, *68*(2), 179-201.
- Electronic Arts. (1998). *SimEarth*. [Computer software]. Walnut Creek, CA: Electronic Arts Inc.
- Feurzeig, W., & Roberts, N. (Eds.). (1999). *Modeling and simulation in science and mathematics education*. New York, Springr-Verlag.
- Fisher, D. (1998). *Mistakes made in the early years teaching students and teachers to create system models*. Paper presented at the 16th International System Dynamics Conference, Quebec City, Canada, July 20-23. [Available on the CC-SUSTAIN Web site: http://www.teleport.com/~sguthrie/cc-stadus.html]
- Ford, A. (1999). *Modeling the environment: An introduction to System Dynamics modeling of environmental systems*. Washington, D.C.: Island Press.
- Forrester, J. W. (1961). *Industrial dynamics*. New York: John Wiley & Sons, Inc.
- Forrester, J. W. (1968). *Principles of systems (Second preliminary edition)*. Cambridge, MA: Wright-Allen Press, Inc.
- Forrester, J. W. (1969). *Urban dynamics*. Cambridge, MA: The M.I.T. Press.
- Forrester, J. W. (1971). *World dynamics*. Cambridge, MA: Wright-Allen Press, Inc.
- Hannon, B., & Ruth, M. (1994). *Dynamic modeling*. New York: Springer Verlag.

- High Performance Systems. (2000a). *ithink*. [Computer software]. Hanover, NH: High Performance Systems.
- High Performance Systems. (2000b). *Stella 6*. [Computer software]. Hanover, NH: High Performance Systems.
- Imagine That. (2000). *Extend*. [Computer software]. San Jose, CA: Imagine That, Inc.
- Maani, K. E., & Cavana, R. Y. (2000). *Systems thinking and modeling: Understanding change and complexity*. Auckland, New Zealand: Pearson Education New Zealand.
- Macromedia. (1999). *Authorware 4.* [Computer software]. San Francisco, CA: Macromedia.
- Mandinach, E. B., & Cline, H. F. (1994). *Classroom dynamics: Implementing a technology-based learning environment*. Hillsdale, NJ, Lawrence Erlbaum.
- Mandinach, E. B., & Cline, H. F. (1996). Classroom dynamics: The impact of a technology-based curriculum innovation on teaching and learning. *Journal of Educational Computing Research, 14*(1), 83-102.
- PowerSim. (1999). *PowerSim*. [Computer software]. Bergen, Norway: PowerSim.
- Pugh, A. L. (1983). *DYNAMO user's manual: including DYNAMO II/370, DYNAMO II/F, DYNAMO III/370, DYNAMO III/F, DYNAMO III/F+, DYNAMO IV/370, and Gaming DYNAMO. 6th ed*. Cambridge, MA: M.I.T. Press.
- Pugh-Roberts Associates. (1982a). *Micro-DYNAMO*. [Computer software]. Reading, MA: Addison-Wesley.
- Pugh-Roberts Associates. (1982b). *User Guide and Reference Manual for Micro-DYNAMO System Dynamics Modeling Language*. Reading, MA, Addison-Wesley.
- Quinn, J., & Alessi, S. (1994). The effects of simulation complexity and hypothesis-generation strategy on learning. *Journal of Research on Computing in Education*, *27*(1), 75-91.
- Roberts, N., Anderson, D., Deal, R., Garet, M., & Shaffer, W. (1983). *Computer Simulation: A System Dynamics Modeling Approach*. Reading, MA: Addison-Wesley.
- Singhanayok, C., & Hooper, S. (1998). The effects of cooperative learning and learner control on students' achievement, option selections, and attitudes. *Educational Technology Research & Development*, *46*(2), 17-33.
- Soloway, E., Pryor, A. Z., Krajcik, J. S., Jackson, S., Stratford, S. J., Wisnudel, M., & Klein, J. (1997). ScienceWare's Model-It: Technology to support authentic science inquiry. *T.H.E. Journal, 25*(3), 54-56.
- Sterman, J. D. (2000). *Business dynamics: Systems thinking and modeling for a complex world*. Boston, McGraw-Hill.
- Ventana Systems. (1999). *Vensim*. [Computer software]. Harvard, MA: Ventana Systems.
- Wenger, E. (1987). *Artificial intelligence and tutoring systems: Computational and cognitive approaches to the communication of knowledge*. Los Altos, CA: Morgan Kaufmann Publishers.
- Zaraza, R., & Fisher, D. (1997). *Introducing System Dynamics into the traditional secondary curriculum: The CC-STADUS project's search for leverage points*. Paper presented at the 15th International System Dynamics Conference, Istanbul, Turkey, August 19-22. [Available on the CC-SUSTAIN Web site: http://www.teleport.com/~sguthrie/cc-stadus.html]
- Zaraza, R., & Joy, T., & Guthrie, S. (1998). *Modeling in the educational environment – Moving from simplicity to complexity*. Paper presented at the 16th International System Dynamics Conference, Quebec City, Canada, July 20-23. [Available on the CC-SUSTAIN Web site: http://www.teleport.com/~sguthrie/cc-stadus.html]

volver