Jl. of Computers in Mathematics and Science Teaching (2002) **21**(4), 313-330

Hybrid Systems Modeling in Learning Science and Technology

TZUR LEVIN AND ILYA LEVIN
Tel Aviv University
Israel
levintzu{ilia}@post.tau.ac.il

The system approach in science and technology education is concerned mainly with continuous systems, whose behavior is described by differential and difference equations. However, in the digital era, the use of discrete systems becomes more and more popular. In particular, the design of hybrid systems—which combines digital and analog (continuous) subsystems—is receiving attention through computer-embedded systems and decision-controlled systems.

This article studies a way to integrate hybrid systems in science and technology classes. It proposes a new class of hybrid systems called Algorithmic Hybrid Systems, which are based on algorithmic notation. An argument in favor of introducing such hybrid systems in science and technology lessons is presented. A method for modeling and simulating hybrid systems using system dynamics simulation software is proposed, and several examples are presented and discussed.

System dynamics is a methodology for studying and managing complex systems (Forrester, 1961). Incorporating constructivist-learning principles (Piaget, 1971), it has been applied to education through a computer-based environment aimed at K-12 classes and beyond (Forrester, 1994). Students study the behavior of simple and complex systems in various subjects, by constructing computer models and running simulations. They acquire system-thinking skills, develop a panoramic multidisciplinary outlook, and learn about specific systems (Chen & Stroup, 1993). The approach has already been applied around the world in science, math, social sciences, business and the humanities. For examples visit the Creative Learning Exchange site at http://www.clexchange.org

The focus of system dynamics in general, and education in particular, has been mainly on continuous systems, whether these are physical, biological, or social systems (Labinaz, Bayoumi & Rudie, 1997). Variables of continuous systems are defined as continuous sets, and modeled by means of differential and difference equations. The behavior of such systems is mainly determined by feedback loops that describe direct and indirect causal relations among elements (Forrester, 1968).

However, in recent years the importance of systems in which continuous and discrete elements interact has been recognized. Particular attention has been given to *hybrid systems,* in which a discrete logical part controls a continuous part of the system (Branicky, 1995). The logical controller is usually modeled as a finite automaton, also called Finite State Machine (FSM). This concept is widely used in computer embedded systems design.

This article studies the potential of hybrid system modeling in the context of learning science and technology. It shows the relevance of hybrid systems to high school education, and describes uses of system dynamics software tools for constructing and exploring hybrid systems in schools.

The main contribution of the article is in introducing a novel approach to the representation of hybrid systems, which is especially suitable for educational needs. This approach is based on the concept of *algorithmic hybrid systems,* in which the controlling part of the system is represented in the form of an algorithm. The algorithmic model of hybrid systems uses fundamental concepts taken from the computer science curriculum (algorithm) on the one hand, and from the classical science curriculum (differential equation) on the other hand.

The article is organized as follows: a section that defines hybrid systems, and describes their relevancy to science and technology education. A section that presents the Algorithmic Hybrid System. This is followed by a section that proposes how to implement the model using system dynamics software (*STELLA*). Two examples are given in the next section. The final section is the conclusion.

HYBRID SYSTEMS

Much recent research has been devoted to explore the mathematical framework for hybrid systems, towards developing a general theory of hybrid systems (Branicky, 1995; Labinaz, Bayoumi, & Rudie, 1997; Mosterman & Biswas, 2000). Nevertheless, the potential role of hybrid systems in science and technology education has not been studied. To do so, a clarification of the nature of such systems is required.

Analog and Digital Systems

Systems are generally described as complex objects whose components are inter-related (Bunge, 1974). By dynamic systems we mean systems wherein processes are developing over time. Dynamic systems may be classified into continuous and discrete, depending on their type of variables. By *analog systems* we mean systems that contain only continuous-valued variables. By *digital system* we mean systems that contains only discrete-valued variables (Branicky, 1995).

Analog systems may be either continuous or discrete in the time domain. Analog systems of continuous-time are usually represented by differential equations. Analog systems of a discrete-time are described by difference equations. When modeled on a discrete-time device, such as the digital computer, differential equations are represented as difference equations, and may be in this sense considered as equivalent (Palm, 1983). In the context of system dynamics, these systems are commonly called *feedback systems*, since the feedback relation between elements of the systems has a crucial effect on the systems' behavior (Forrester, 1968).

Digital systems are either infinite or finite. A well-known infinite digital system model is the Turing machine (Harel, 1993). The Finite State Machine (FSM) as a fundamental model of finite digital systems is commonly used in computer science and digital control engineering (Hopcroft, Motwani & Ullman, 2001; Mano & Dime, 1997; Varshavsky & Pospelov, 1988).

The Hybrid Approach

By *hybrid systems* we mean dynamic systems that are a combination of analog systems and finite digital systems. In a typical hybrid system a digital unit controls the behavior of continuous processes.

The digital subsystem is usually modeled as an FSM, while the analog subsystem is modeled by differential and/or difference equations. The behavior of the digital part of the system can be described using the state transition rules related to the corresponding FSM, while the analog part changes over time as a derivative/integral function. Graphically, the digital part is represented as a state diagram, while the analog part is usually represented as a stock-flow diagram (both types of diagrams will be described later).

The architecture of a hybrid system forms a two-level control structure. On the low level, feedback loops are used for local control as part of the analog system. On the high level, meta-control logic function switches between modes of the analog systems behavior according to a predefined logic of the FSM's state transition rules (Bencze & Franklin, 1995; Branicky, 1995; Mosterman & Biswas, 2000). The properties of analog and digital systems are summarized in Table 1.

Table 1
Properties of the Digital and Analog System

	Digital	*Analog*
Mathematical model	Final state machine Algorithmic state machine	Differential/difference equations
Behavior	State transition according to input and transition rules	Change as a derivative/integral function
Graphical model	State diagram	Stock-flow diagram
Type of control	Event driven control	Local feedback loops

The concept of hybrid system has been known since the early days of dynamic systems theory (Branicky, 1995), but its importance in the modern man–made digital world is still on the rise. Implementations of hybrid systems modeling methods are recognized mainly in the two following contexts:

1. In a local context, in cases when microprocessors control physical and technological processes.
2. In a global context, when human decision making systems may be modeled by means of hybrid techniques.

The first group consists of modern technological real-time systems (Antsaklis & Lemmon, 1998). Cars, robots, cell phones, medical devices, home climate systems, microwaves, computer disk drives, washing machines, intelligent transportation systems, and production lines are all technologies where a digital controller interacts with continuous processes (Branicky, 1995; Johansson, 2000). Most of today's sophisticated controlling systems use the computer as a mean of control, though their full potential is still to be revealed (Kamil & Chui, 1996). The presence of these systems in everyday life—from consumer electronics to traffic control—is constantly growing.

The second group of hybrid systems is an outcome of the interference of man in natural and social processes. In such cases a control policy is applied to natural and social processes, for better or for worse. Environmental issues can be modeled as problems of logical control, and they can therefore be represented as hybrid systems. Similar models may be applied to mixed economies, where market mechanisms intertwine with government and central bank policies. The growing impact of human decisions on the natural environment, and the increasing complexity of modern technological society makes those hybrid entities a key factor in understanding modern reality.

Several examples of such systems are offered by Table 2 (Levin, Levin, & Talis, 2001):

Table 2
Examples of Hybrid Systems

Subject matter	***Digital controlling subsystem***	***Analog controlled subsystem***
Economics	Government and central bank fiscal policy	Market mechanism
Geography	City council decisions	Urban development
Ecology	Hunting regulations	Prey- predator relations
Medicine	Drug prescription	Drug absorption in the body
Transportation	Rules of traffic	Traffic flow
Applied ethics	Affirmative action policy	Mobility of minorities in society

Hybrid Systems in Science and Technology Education

In the system dynamics approach, students create computer models and run simulations to solve given problems, and thus construct their knowledge and improve their mental models about natural and technological system (Forrester, 1994). A focus on constructing *hybrid* models of systems may assist learning in this approach because:

1. It provides a systematic methodology for system design. When designing hybrid systems students will be asked to:
   a. design the analog (controlled) system (stock-flow);
   b. design the digital (controlling) system (FSM); and
   c. explore the interaction between the two subsystems (Bencze & Franklin, 1995).
2. It covers a plurality of systems including both natural and technological components, and especially systems that contain microprocessors.
3. It supports a problem-based approach to teaching, which is typical to the system dynamics methodology (Andersen & Richardson, 1980). Control tasks are naturally formulated and studied with hybrid models.

4. It provides a meaningful context for teaching fundamental concepts from discrete math, computer science and digital design. These concepts are now mainly taught in engineering and computer science classes and therefore they are limited to a small group of professionals. Nevertheless, the influence of these concepts on the modern world should make them part of general education, in the framework of science and technology education for all. Students with different interests will find contexts in which the construction of a digital controller will be meaningful activity for them—be it a physical, medical, environmental, economical, or other.

The Algorithmic Hybrid System Model

There is a plurality of models representing hybrid systems (Branicky, 1995). This article proposes to adopt a special type of model, based on ideas taken from computer engineering education, and suitable for educational needs (Baranov, 1994). This model divides the system into two interacting parts—one controlling and one operational. The controlling part is represented as an Algorithmic State Machine (ASM), which is a specific interpretation of FSM appropriate to the school curriculum. The operational unit is any set of differential, or difference, equations.

In the dynamic interaction between the parts, the operational unit sends a certain signal to the controller and receives an instruction, which guides it for the next operation. Obviously, inputs of the analog operational part of the system are outputs of the digital controller and vice versa (Figure 1).

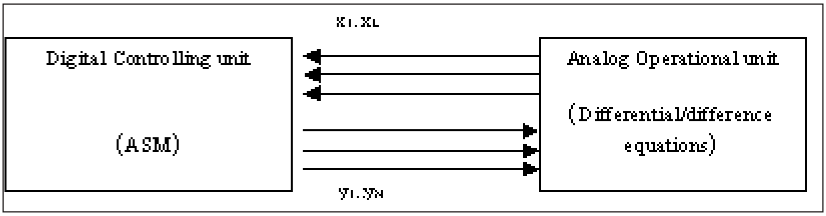

Figure 1. A general structure of a hybrid system

We will call this type of hybrid systems *algorithmic hybrid systems* and define it as follows:

An Algorithmic Hybrid System is a Hybrid Dynamic System whose controlling part is described in a form of the Algorithmic State Machine.

An ASM as a form of controller description is very close to the FSM, and both descriptions (ASM and FSM) are mutually translatable (Baranov, 1994). In educational context, the ASM has a number of advantages, which will be addressed later. Let us define the ASM based controller for the hybrid system.

Let a *micro-operation* be an elementary action in the analog part of the system and let Y = $\{y_1,..,y_n\}$ be a set of micro-operations. These micro-operations are induced by binary signals $y_1,..,y_n$ form the controlling part of the system. We propose considering a micro-operation as a signal authorizing switching on a certain “flow” within the analog part of the system.

Let us present the simple example of a water-level controller. Given is a water-tank with a tap to fill with water and a drain to empty the tank, and given also is a random outflow function. The task is to design an ASM to keep a constant high water level (Figure 2).

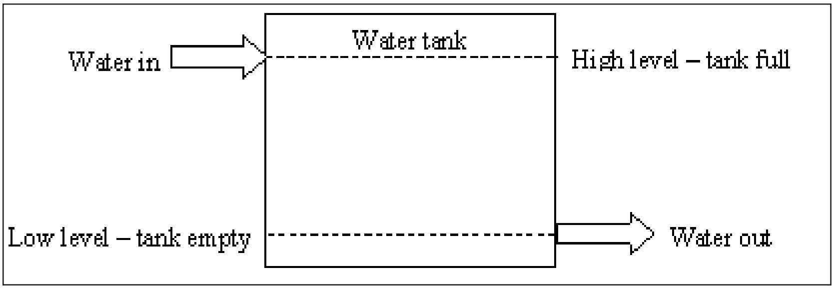

Figure 2. Water level control

In our example two micro-operations are defined: y_1 means to open the inflow tap and y_2 means to close the inflow tap.

A sequence of micro-operations is determined by transition functions, that is, Boolean functions of binary variables $x_1,..,x_L$. Being input variables of the digital controller, these variables are binary predicates defined on the set of “stock” values of the analog part of the system. In our example: x_1= (water level £ high level); x_2= (water level ^{3} low level).

ASMs are usually presented in the form of a graph consisting of an initial vertex, a final vertex and a finite set of operator vertexes and conditional vertexes. One of the predicates $x_1,..,x_L$ is written in each conditional vertex. A micro-operation is written in each operator vertex. A sample of ASM graph is shown in Figure 3.

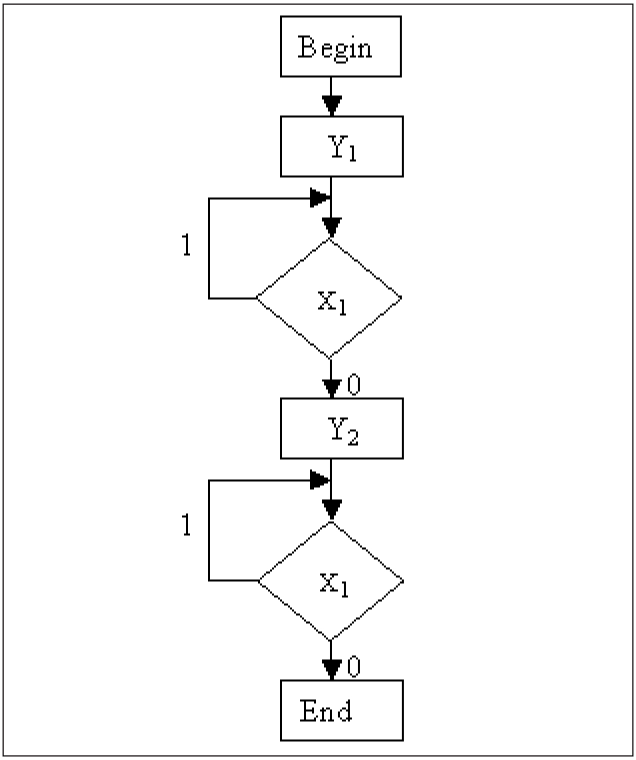

Figure 3. The ASM for water tank example

The ASM describes the following algorithm: Open the tap. Keep it open until the high level is reached (x_2=1). Then close the tap, and keep waiting till water flows out of the tank. Then open the tap again and so on.

IMPLEMENTATION OF ALGORITHMIC HYBRID MODEL

Computer simulation is a major tool in system dynamics based teaching. A close look at the way modeling and simulation are applied reveals different approaches in terms of (a) software platform; (b) types of models; and (c) the role students play in the models construction (Sterman, 1991; Alessi, 2000, Maier & Grobler, 2000).

STELLA as Platform

In terms of software platform, hybrid modeling presents a unique challenge since it requires the construction of three components: (a) the analog system, (b) the digital system, and (c) the interaction between them. For didactical reasons the use of the same software for both the digital and the analog parts is preferred. The following section shows how both the analog and the digital parts of the system can be modeled and integrated in *STELLA* (High Performance Systems, 1985-2001).

STELLA is software for systems dynamics modeling, widely used in K-12 classes around the world. It uses an iconic language of stocks-flow diagrams, to which connectors; converters, feedback loops, and other elements are added to create models of complex systems. After the model has been completed, students run the simulation to explore the graphical behavior of key elements in the system.

Analog Part Modeling

The basic vocabulary of *STELLA* is that of stocks and flows, followed by converters and connectors. Stocks are graphically represented as rectangles, and can be of four types: reservoirs, conveyors, queues, and ovens. Flows are pipes with spigots through which a flow is moving into and out of stocks. Converters and connectors are used to modify variables, to connect elements and to let them influence one another (e.g., through feedback loops).

As an example the water tank model is shown in Figure 4. The shapes are presented on the screen by dragging and dropping, and values are given by filling dialog boxes. The resulting models are presented in the form of the set of differential equations whose behavior can be evaluated by running a simulation.

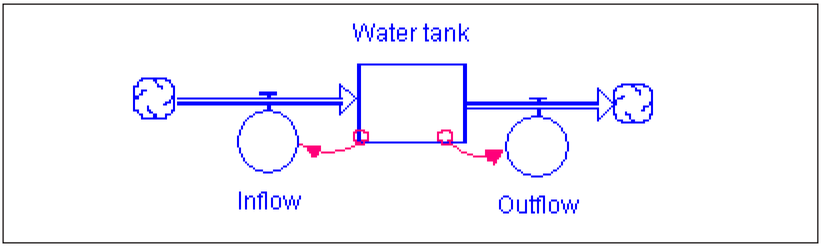

Figure 4. A stock-flow description of a water tank model

The stock-flow modeling language combines a user-friendly interface with a power to simulate complex continuous systems. The vocabulary of this language can be manipulated to enable simulation of digital components.

Modeling the Algorithmic State Machine in *STELLA*

Starting with an ASM description of the controlling part we transform it into an FSM description. The structure scheme of the FSM is shown in Figure 5. The structure scheme of the digital controlling system consists of two elements:

- a combinational scheme, which receives inputs and calculates outputs;
- a memory register, which stores the current state of the systems.

The digital controller receives input signals from the analog system, and executes combinational logic functions to determine the output and the next state of the systems.

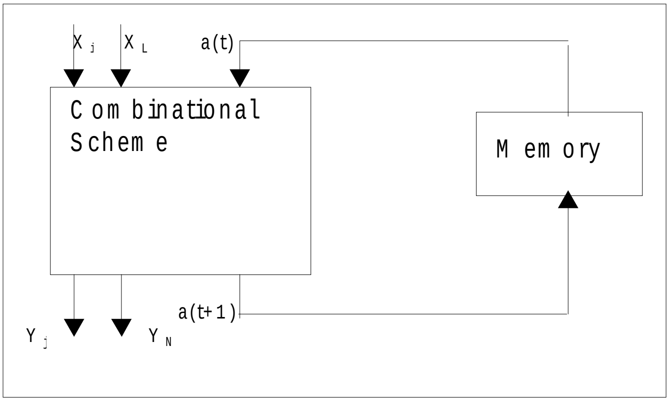

Figure 5. Structure scheme of the controlling

Each of the elements of the controlling unit may be implemented in *STELLA* as follows. Our proposal for implementing the combinational scheme in *STELLA* is by using a logical device called multiplexer. The multiplexer is a combinational logic device, which receives several input variables and selects one of them to be its output (Mano & Dime, 1997).

The selection is controlled by a set of logical conditions. It will now be demonstrated how a multiplexer can be constructed using *STELLA*'s connectors and convectors.

In our example the multiplexer receives two inputs, x_1 and x_2, from the controlled sub-system, and the current state stored in the memory - a_1 or a_2. The multiplexer returns the new state of the system – a_1 or a_2 - according to the values of x_1, x_2 and the current state.

The multiplexer-based implementation of the combinational scheme of the controller can be represented in *STELLA* as tree-graph with converters as nodes. The nodes on the lowest level of the graph represent values of input variables. The graph executes a function of selection between the variables based on conditional statements (if-then). The output of the multiplexer affects a corresponding flow (Figure 6).

Memory is implemented in *STELLA* by means of a stock with two flows. Each state has a numerical value (a_1=1, a_2=2, etc.), and the value of the current state is kept in the stock (Figure 6). With each "move" of the controlling systems, the value of stock nulls through the outflow, and a new value is added through the inflow, to be used as the value for the next calculation. Thus, the effect of delay, which is essential to the FSM, is achieved.

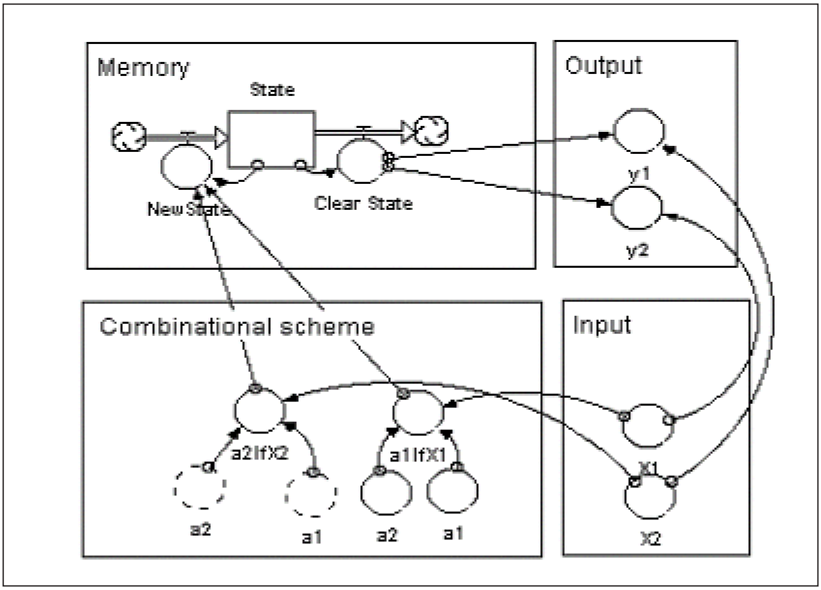

Figure 6. FSM implementation in *STELLA*

Modular Approach to Students' Role

The role of students is described on a scale between gaming-oriented and modeling-oriented approaches (Maier & Grobler, 2000). In the first extreme, students are engaged in playing with existing models as in simulation games in the tradition of *SimCity*; on the second extreme, they construct models from scratch.

The separation between the controlling and the controlled units in our hybrid approach enables a modular approach. The continuous controlled subsystem may be either given or modeled by students. The controlling functions may be achieved manually, as in games, or automatically by a digital controlling unit designed by students (Table 3).

Table 3
Students' Activities

	Controlling unit	Controlled unit
Playing	Manual control	Given
Automatic control design	ASM implementation	Given
Whole system design	ASM implementation	Stock flow modeling

EXAMPLES

Artificial Climate Control

A simulation of an artificial climate system is an example associated with both physics and technology education. The physical process is that of air cooling and heating. The technological aspect concerns the way air-conditioning functions. To gain basic understanding of the subject we simplify both aspects of the systems. First, we model the physical process as change in temperature (though the real flow is, of course, of energy, the temperature being a side effect). Second, we focus solely on the controlling element of the air-conditioning system, ignoring the way cold and hot air is produced. Further improvement of the accuracy and richness of the model may be achieved at a more advanced stage of the teaching.

The problem presented to the students is a typical control problem:

> *Create a digital thermostat that will regulate the temperature of a room around the value of a thermostat.*

The traditional way to model this system is by using a proportional-integral controller, which turns a radiator on and off. An alternative approach is based on a digital controller, represented as ASM. The controller will monitor the range of the room temperature, and will turn the radiator on and off according to the state of system. The system is described by its continuous and discrete subsystems, input and output vectors and a state vector (Figure 7):

Input vector: x_1 = (Room Temp ^{3} Thermostat+2); x_2 = (Room Temp £ Thermostat-2)

Output vector: y_1 means turn on the radiator; y_2 means turn off the radiator.

System states: a_1 means radiator on, while a_2 means radiator off.

The graph of the room temperature shows its regulated behavior over time (Figure 8). The simulation may be used for what-if scenarios, to evaluate the effects of the level of insulation, and to explore changes in the thermostat settings. More advanced models may describe the flow in the physical system in terms of energy changes and include more features of control such as humidity and smoke control.

Drug Prescriptions

Medical treatment can be viewed as adopting control strategies to regulate the behavior of natural processes. Consider this example: a patient is to receive a dangerous drug. The drug enters the blood stream and then moves on to the stomach, where its effectiveness can be measured. A minimum concentration of the drug in the stomach is required to have therapeutic value. However a high concentration of the drug might be dangerous and even kill the patient.

Students are asked to model the process of drug absorption in the blood, and to design a digital control unit to regulate the consumption of the drug in real time. The controller has to monitor the amount of drug in the patient's body, and to determine whether to give the patient an additional dosage. The tricky point in this exercise is the issue of delay. Since the drug has to go first through the blood stream and only then reaches the stomach, it takes time until control decisions are felt in the body.

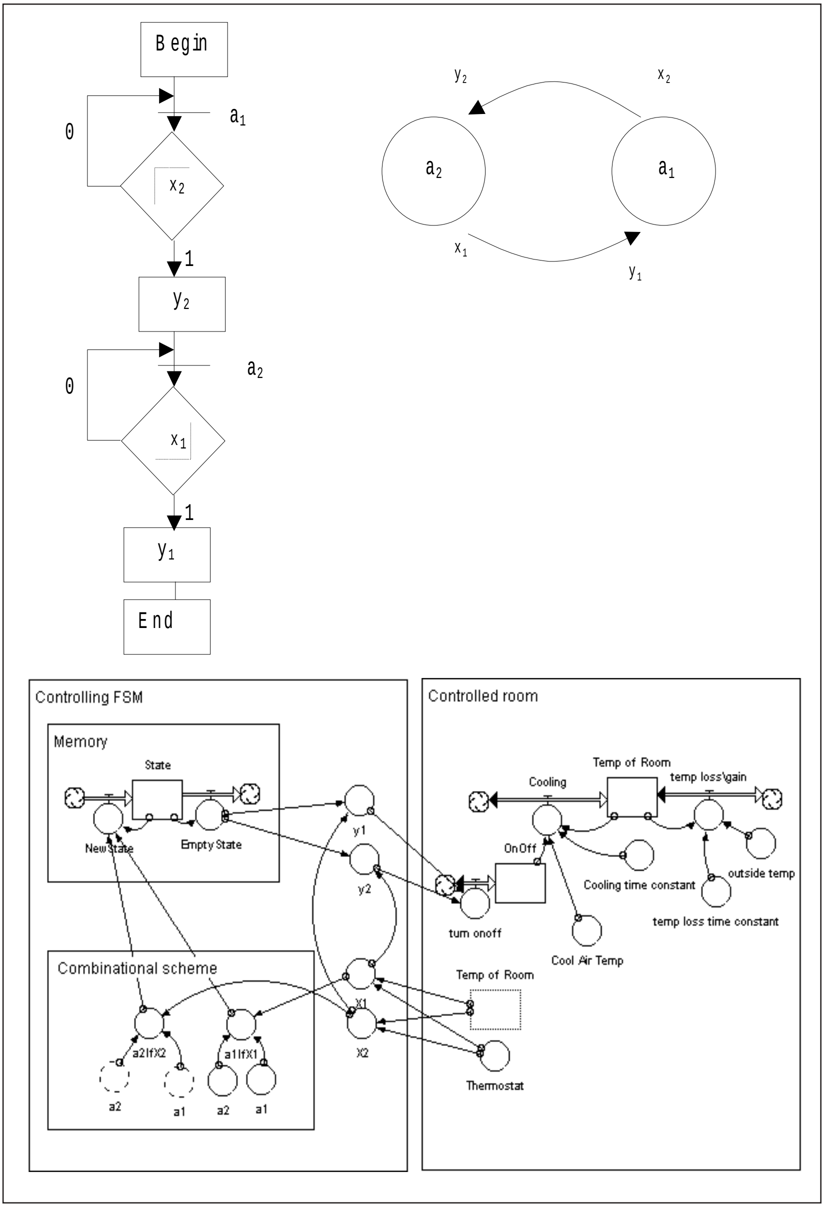

Figure 7. ASM (top left), FSM (top right) and implementation in *STELLA* of climate system

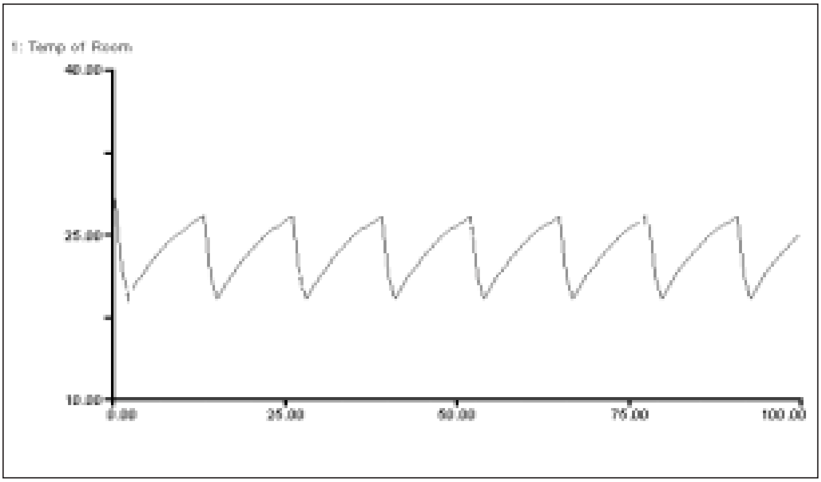

Figure 8. Room temperature graph

This problem may be given to students in biology, medical and para-medical classes, as well as control students in technology and engineering classes. The problem can be stated as follows:

> *Design an automatic controlling mechanism to treat a patient using a dangerous drug. The concentration of the drug in the stomach should only be values between the minimum effective and the toxic levels.*

Students are expected to design the physiological model first, and then to create the controlling element as an algorithmic state machine. Both the continuous and the discrete sub-systems are to be implemented in *STELLA* as shown in Figure 9.

Input vector:

x_1 = (Blood concentration £ Minimum therapeutic concentration+0.05)

x_2 = (Blood concentration ^{3} toxic concentration-0.25)

Output vector: y_1 means set the dosage to 20 mg; y_2 means set the dosage to 5 mg.

System states: a_1 means 20 mg dosage is being taken; a_2 means 5 mg dosage is being taken.

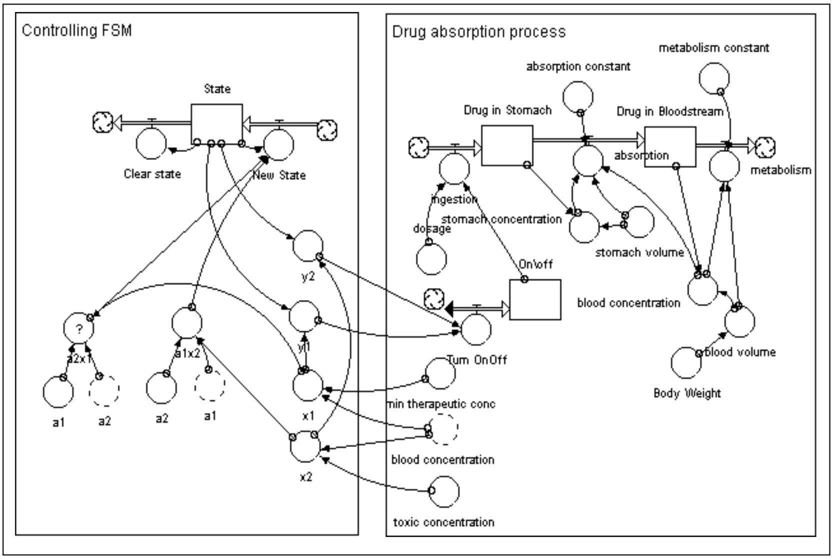

Figure 9. Drug prescription implementation in *STELLA*

The graph in Figure 10 describes the drug concentration in the blood over time for the duration of the simulation.

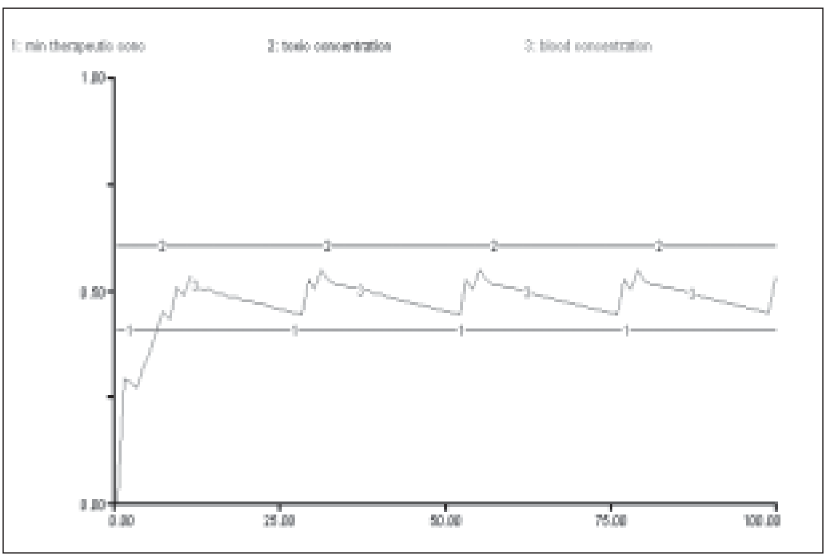

Figure 10. Concentration graph

SUMMARY AND RESEARCH AGENDA

This article proposes a novel approach for teaching systems in science and technology education. The approach is based on the concept of the Algorithmic Hybrid System, which is a Hybrid System with an algorithmic-based control part. In spite of the great advance made in the theory of hybrid systems, efficient teaching oriented models of hybrid systems have not been developed. We have tried to fill this hiatus.

The Algorithmic Hybrid Systems approach has two clear advantages in the educational context: (a) it is based on the concept of the algorithm, which is familiar to many science and technology students through computer science studies, and (b) it offers a systematic methodology for solving problems

Future research on hybrid system modeling in education should follow two directions:

1. On the theoretical level, more examples need to be constructed and analyzed in diverse subject matters.
2. On the didactical level, empirical research on the pedagogical aspects of using algorithmic notation in coordination with stock-flow notation is required.

References

Alessi, S. (2000). Designing educational support in system-dynamics-based interactive learning environments. *Simulation & Gaming, 31*(2), 178-196.

Andersen, D.F. & Richardson, G. (1980). Toward a pedagogy of system dynamics. *TIMS Studies in the Management Sciences, 14*, 91-106.

Antsaklis, P.J & Lemmon, M.D. (1998) Introduction to the special issue on hybrid systems. *Journal of Discrete Event Dynamic Systems, 8*(2), 101-103.

Baranov, S. (1994). *Logic synthesis for control automata*. Dordrecht/Boston/London: Kluwer Academic.

Bencze, W.J., & Franklin, G.F. (1995). A separation principle for hybrid control system design. *IEEE Control Systems Magazine, 15*(2), 80-84.

Branicky, M.(1995). *Studies in hybrid systems: Modeling, analysis, and control*. Unpublished doctoral dissertation, MIT, Cambridge.

Bunge, M. (1974). *Treatise on basic philosophy. Volume 4: A world of systems*. Dordrecht: Reidel Publishing.

Chen, D., & Stroup, W. (1993). General system theory: Toward a conceptual framework for science and technology education for all. *Journal of Science Education and Technology, 2*(3), 447-459.

Forrester, W.J. (1961*) Industrial dynamics*. Cambridge, MA: MIT Press.

Forrester, W.J. (1968). *Principles of systems* (2nd ed). Cambridge, MA: Wright-Allen Press.

Forrester, W.J. (1994). *Learning through system dynamics as preparation to the 21st century*. Keynote Address for System Thinking and Dynamic Modeling Conference for K-12 Education. Concord Academy. Concord, MA.

Harel, D. (1993). *Algorithmics: The spirit of computing* (2nd ed). Reading, MA: Addison-Wesley.

Hopcroft J.E, Motwani, R., & Ullman, J.D. (2001): *Introduction to automata theory, languages, and computation.* (2nd ed). New York: Addison-Wesley.

Johansson, K. (2000). *Hybrid systems. Spring 2000 Lecture notes.* Berkeley. [Online]. Available: http://robotics.eecs.berkeley.edu/~johans/ee291e.html

Kamil, A., & Chui H.W. (1996). Hybrid control systems and optimal control. *Surprise, 4.* [Online]. Available:http://www.doc.ic.ac.uk/~nd/surprise_96/journal/vol4/ahak/report.html

Labinaz, G., Bayoumi, M.M., & Rudie, K. (1997). A survey of modeling and control of hybrid systems. *Annual Reviews in Control, 21,* 79-92

Levin T., Levin, I., & Talis, V. (2001). *System dynamics learning through separation of a control unit*. Paper presented to the international Patt-11 conference. Harlem, Netherlands.

Maier, F.H., & Grobler, A. (2000). What are we talking about?—Taxonomy of computer simulations to support learning. *System Dynamics Review, 16*(2), 135-148.

Mano, M.M., & Dime, C.R. (1997). *Logic and Computer Design Fundamentals*. New Jersey: Prentice-Hall.

Mosterman, P.J., & Biswas, G.B. (2000). A comprehensive methodology for building hybrid models of physical systems. *Artificial Intelligence, 121*, 171–209.

Palm, W. (1983). *Modeling, analysis and control of dynamic systems*. New York: John Wiley & Sons.

Piaget, J. (1971). *Biology and knowledge: An essay on the relations between organic regulations and cognitive processes*. Chicago: University of Chicago Press.

Sterman, J.D. (1991). A skeptic's guide to computer models. In G.O. Barney, et al. (Eds.), *Managing a nation: The microcomputer software catalog,* 209-229. Boulder, CO: Westview Press.

Varshavsky, V.I., & Pospelov, P.A. (1988). *Puppets without strings*. Trans. by A. Dandarov. Moscow: Mir Publishers.