Integrating Hybrid Modeling with System Dynamics

Tzur Levin and Ilya Levin
Tel Aviv University, Schools of Education
Ramat Aviv, Israel
Tel: (972)-3-6407109, Fax: (972)-3-6407752
Email: levintzu@post.tau.ac.il

Abstract

This paper explores how hybrid modeling may be combined with the system dynamics methodology in order to serve didactical and communicative goals. It describes the architecture of hybrid models and argues for its compatibility with the system dynamics outlook. The paper specifies cases suitable for hybrid modeling and presents examples to demonstrate its pedagogical value.

1. Introduction

In recent years the engineering community has given growing attention to models of hybrid systems in which continuous and discrete variables coexist. Several works have proposed a conceptual framework for such systems (Asarin et. al, 2000; Maler, 2001; Gupta et.al, 1998), while others studied specific models (Breedveld, 1999; Mosterman & Biswas, 2000). This interest is due to the suitability of hybrid models to describe sophisticated control systems, in particular, computer-embedded systems.

Hybrid system theory connects two models of change, one described by continuous differential equations and the other by discrete logical transitions. Traditionally, differential equations have been used to describe "real time" behavior of natural processes (Lebinaz et al., 1997). Discrete transitions, widely applied in computer science, have been used to depict the "logical-time" behavior of automatic machines (Maler, 2003). Mixed systems have usually been reduced to "pure" forms, either continuous or discrete (Branicky, 1995).

The segregation between the continuous and the discrete modes of dynamics dominated the social sciences as well. It is a major point of dispute between the cybernetic and servomechanistic threads (Richardson, 1991). The cybernetic thread explored the social world on the level of messages and events, while the servomechanistic thread moved "one step back from events" to view a smoother picture of social change (Forrester, 1961). Following the hybrid approach, our research explores how the two threads might complement rather than contradict one another.

Our motivation is to enrich system dynamics (SD) education and suggest new ways of interaction with SD models. This is important because SD addresses a wide range of audiences with diverse backgrounds, from elementary school children through university graduate students to senior management staff. Research has shown that system-thinking skills are scarce even among elite students (Sweeney & Sterman, 2000; Sterman & Sweeney, 2002; Ossimitz, 2002). In particular, stock-flow reasoning, which is a basic building block of continuous systems, is far from intuitive. Attempts to resolve this difficulty by using qualitative causal diagrams were found problematic (Richardson, 1986, 1997). We believe hybrid models are a reasonable approximation to continuous models but are easier to comprehend. Thus, they offer system dynamics educators a "good tradeoff between real-world relevance and model complexity" (Maler, 2001).

The paper is organized as follows: First the hybrid architecture is introduced, and then an argument for its compatibility with the system dynamics methodology is given. The last section classifies hybrid models along pragmatic criteria and presents examples for their use.

2. The Architecture of Hybrid Systems

The hybrid system consists of a two-level control structure (Figure 1). On the low level, differential equations modeled as stock-flow diagrams form a continuous system. On the high level, logical functions, modeled as a finite state machine (FSM), constitute the discrete part (Levin & Levin, 2002). The two levels are connected through a two-way channel of communication: a binary vector of information ($x_1..x_L$) is sent from the continuous to the discrete part, and a binary vector of instructions

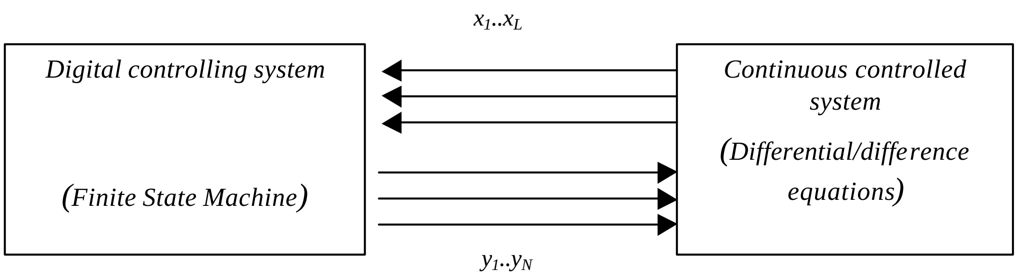


Figure 1. *A general structure of a hybrid system*

($y_1..y_N$) is sent in the opposite direction.

FSM is a fundamental model in computer science (Hillis, 1998). It consists of a memory element, stored as the state of the machine, and Boolean functions expressing the transition rules between states. In the hybrid architecture, transitions are based on the computations of the input vector received from the continuous system. In turn, the FSM sends instructions to switch between modes of the continuous system, a mode being a distinct continuous dynamic law (Asarin et. al, 2000). Thus the overall behavior is that of discrete transitions between continuous modes.

3. Compatibility with System Dynamics

The canonical texts of SD discourage the use of discrete variables. Though Forrester (1961) stated that discreteness is not in principle incompatible with system dynamics, it is considered a bad modeling habit. There are several reasons for this attitude.

First, discrete variables are not compatible with the continuous nature of social reality (Forrester, 1961; Sterman, 2000). System dynamics views social phenomena as an aggregation of particular events, messages and decisions into gradual modification represented by continuous flows. *Second*, continuous dynamics is considered easier to comprehend than discrete dynamics (Forrester, 1961). Sterman (2000) explicitly states that “conditional statements such as IF..THEN..ELSE are more difficult to understand, especially when conditions are complex or nested with others”. *Third*, many model builders, especially those with a strong computer science background, tend to overstress the discontinuities of real situations (Forrester, 1961). To eliminate this bias a correction emphasizing continuity is required.

The objections to the hybrid approach are not unique to the system dynamicists. Other practitioners of the control community consider logical elements as “second class citizens” in a realm ruled by elegant smooth algebraic functions (Asarin et al., 2001). The way the objections are met in the engineering domain is relevant to SD as well.

To begin with, the hybrid approach generally agrees with the continuous account of dynamics: the discrete jumps are between continuous modes of change. Nevertheless, it refuses to accept continuousness as an ontological dogma, and applies pragmatic criteria for employing discrete variables. It hereby follows the steps of the rational-critical school in the philosophy of science, who argued that the success of the 18th century's continuous paradigm of natural science need not imply a dogmatic acceptance of calculus as the only modeling language (Bunge, 1974).

On the psychological level, intuition contradicts the assumption that continuity is easier to comprehend. On the contrary, we hypothesize that when using appropriate notation, like that of a FSM, discrete dynamic will prevail in assisting understanding. For instance, we predict that given a continuous process, a discrete control mechanism described by FSM will be more communicative and easy to learn than its (somehow) equivalent stock-flow controller. We are developing research tools to examine this hypothesis.

As to the methodological issue, hybrid models force a dual perspective of chance. Rather than encouraging fixation of bias, it educates to view change from diverse paradigms. The question, therefore, is not whether to use hybrid modeling but when.

4. Difference from traditional SD models

To some SD practitioners, the hybrid approach may not seem novel. It may look like merely a new name for traditional models combining stock-flow structures with logical statements. Indeed, FSM is equivalent to IF..THEN..ELSE statements, in a sense that each any FSM may be translated to an IF statement and vice versa. They represent the same mathematical model using different notations. Nevertheless, when considered as tools for assisting thought, the difference between the notations is significant.

First, FSM notation is more suitable for describing complex cases of logical computation. Compare the simple 2-states FSM described in Figure 2 to the 4-states FSM in Figure 5. The first may be easily translated to IF structure, while the second leads to an unfriendly nested structure. The ability to handle complexity is one of the reasons that the engineering community commonly uses the FSM notation to describe systems with logical control-dominated-architecture.

Furthermore, the FSM notation represents a declarative mode of thinking as opposed to the procedural nature of IF statements. The first emphasizes thinking about rules, while the second presents an algorithm for action. Part of our research tries to reveal cases in which students prefer to use declarative to procedural notations. The design of SD models may be among these cases, given the declarative characteristics of stock-flow maps.

5. Pragmatic Classification of Hybrid Systems

We present a pragmatic classification of hybrid models from a pedagogical perspective. So far we identified the advantages of using hybrid models in the following contexts:

1. Presenting simple oscillating behavior
2. Describing behavior of multivariable dependence
3. Directing focus on events
4. Designing meaningful mode of interaction with models.

Below we describe each case and provide examples. The examples are not of running models, since the SD modeling tools do not support elegant notation for FSM design. Yet the graphs below describe authentic behavior of the models, based on a technique for constructing FSMs in STELLA detailed elsewhere (Levin & Levin, 2002).

5.1 Presenting Simple Oscillation

First-order continuous systems cannot oscillate (Sterman, 2000). Therefore to present oscillation, the construction of a second-order system (or more) is required. Alternatively, we may include discrete variables in a first-order system and launch a new time scale to create delay and therefore oscillation. Thus, hybrid models offer a simple perspective on the phenomena of oscillation, especially for audiences with a weak mathematical background.

Consider the dynamics of the love between Romeo and Juliet (Strogatz, 1988; Radizki, 1993). With J being a stock of Juliet's love and R being Romeo's love stock, a and b being parameters where a*b<0, an harmonic oscillation behavior is described by the following set of equations:

$$\begin{cases} dR/dt=a*J \\ dJ/dt=b*R \end{cases}$$

Using a hybrid model we may oscillate Romeo's love while Juliet's love remains a positive constant. In this case Romeo's love changes in the following pattern: it grows in proportion to Juliet's love till twice its value. Then it starts declining at the same pace, and switches back to growth once reaching the value of half Juliet's love (Figure 2). Figure 3 describes the rules of Romeo's behavior as transitions between states in the FSM, and the equations of the stock-flow part of the model.

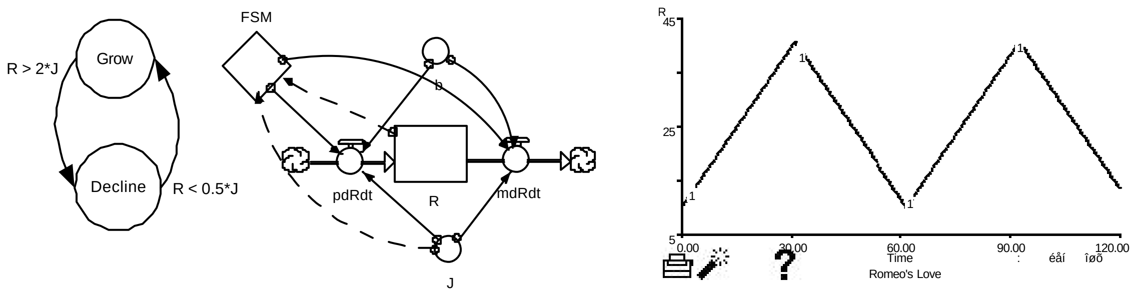


Figure 2 *Hybrid model of Romeo's love: a finite state machine, the continuous model and the linear oscillation*

State / Input	State 0	State 1
R>2*J	State 1	-
F<2*J	-	State 0

*R (t) = R (t - dt) + (pdRdt_2 - mdRdt_2) * dt*
INIT R= 10; b= 0.05; J = 20
*pdRdt_2 = if State=0 then b_2*J else 0*
*mdRdt_2 = If State=1 then b_2*J else 0*

Figure 3 the transition table of Romeo and the equations of the model

Modification of the model leads to diverse oscillations (Figure 3). We may assume that the thresholds of Romeo's love, as a function of Juliet's love, change over time. If they converge towards the value of Juliet's love, it is a damped oscillation. If the thresholds constantly increase, the oscillation has growing amplitude. The case of random change of thresholds is interesting because it creates unpredictable oscillation. Back to the semantics of the love metaphor, we may say that living together leads towards equilibrium, living apart increases the momentum, but the moon often has the

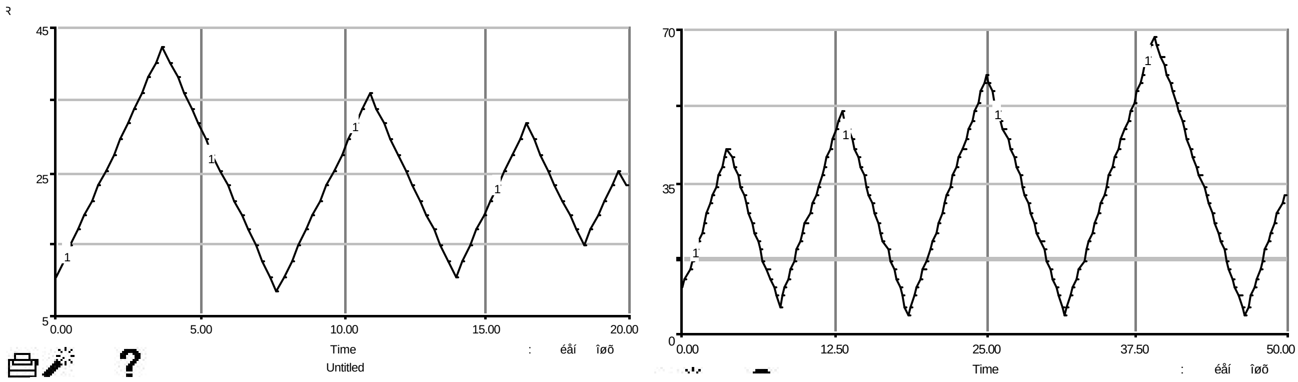


Figure 4 Types of oscillations

last word in matters of the heart.

5.2 Multivariable Dependence

Consider a bath system. Given any initial conditions of a bath equipped with a hot water faucet and a drain exit, the goal is to maintain the water volume around 100 liters and the temperature around 40^0c. In this case the inflow and outflow of water needs to be a function of two variables: the temperature and the volume. Since a two-dimensional graphical description of the function is impossible, the simplest continuous solution is a linear combination of the volume and temperature gaps. Hybrid modeling may be more instructive. FSM with 4 states switches between modes of the system according to input and predefined rules (Figure 5). A description of the states is given in the table below (Table 2).

Faucet / Drain exit	Open	Closed
Open	State 1	State 2
Closed	State 3	State 4

Table 2 The 4 states

The FSM receives the following Boolean input values of temperature and volume of the water: "Hot"=(Temp>45^0); "Cold"=(Temp<45^0); "Full"=(Volume>110); "Empty"=(Volume<90). Figure 5 shows the transition function between states based on the input.

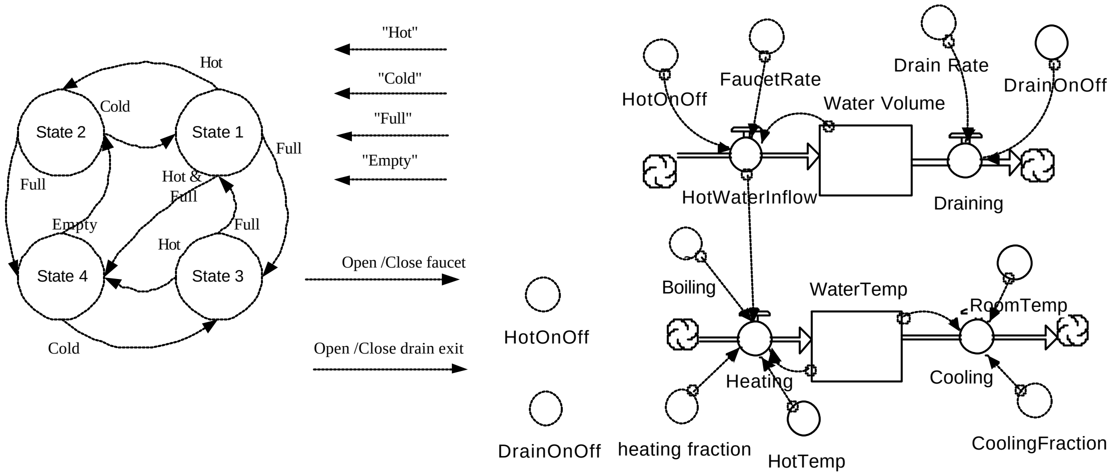
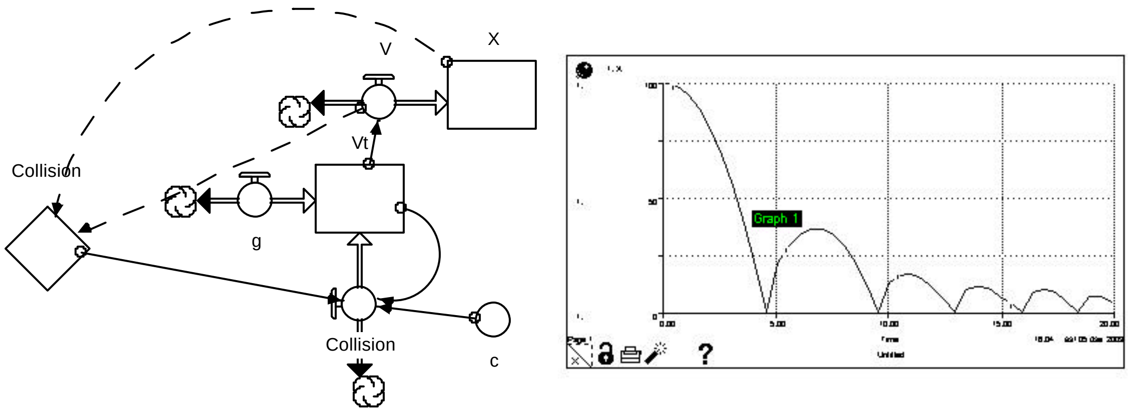


Figure 5 Bath heating system

5.3 Focus on Events

To focus on a certain detail in the system, we raise its level of abstraction (Lacoste-Julien, 2002). With hybrid modeling we do this by assigning discrete values to the desired element, thus changing its time scale compared to the rest of the system.

Consider a hybrid description of a bouncing ball. According to the laws of mechanics, the movement of the ball is continuous. Even the collision with the ground is a gradual deformation of the ball, which eventually makes it bounce. Nevertheless, when we describe the movement of the ball in the air with a stock-flow model, we prefer to emphasize the collision by mean of an instantaneous description (Figure 6). The equations of the model are given below.



Vt(t) = Vt(t - dt) + (g_ + Collision) * dt; INIT Vt = 0
Collision = ((-Vt-Vt*c_)*Collision_Occures)/dt
X(t) = X(t - dt) + (V) * dt; INIT X = 100
V = Vt ; c_ = 2 ; g_ = -9.8
Collision_Occures = if (X=0 and V<=0) then 1 else 0

Figure 6 Hybrid model of bouncing ball

5.4 Meaningful Interaction

There are several possible modes of interaction with models (Alessi, 2000). A user may construct models from scratch, play with existing simulation or do a half-structured activity in between. An interaction suitable for hybrid modeling is to design a real time decision-making algorithm, using the FSM notation, to a given continuous process. This activity may have didactical value since it combines analytic and synthetic modes of thinking (Levin & Lieberman, 2000).

Back to the story of Romeo and Juliet, let us assume their relations result in continuous harmonic oscillation, with only one fourth of the time both being in love simultaneously (Figure 7). Now let us introduce a third player into the model –

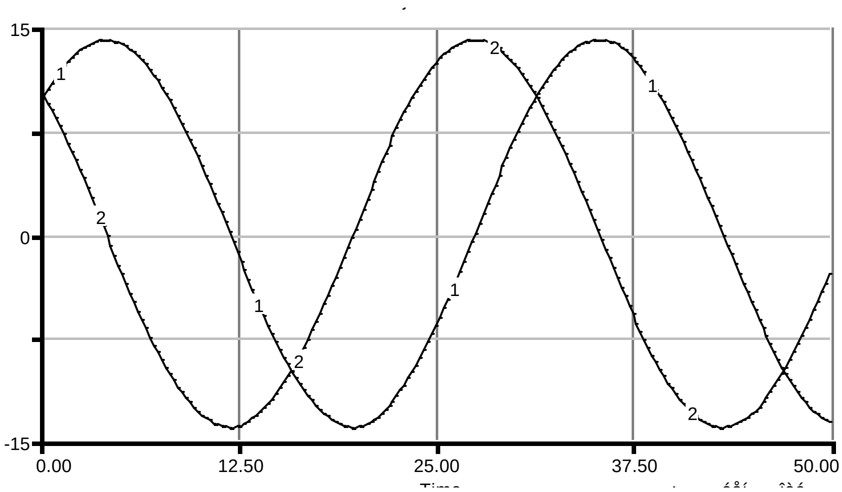


Figure 7 Harmonic oscillation of Romeo and Juliet

Lorenzo, the “love consultant”. What can Lorenzo do to increase the poor lovers’ happiness? He may, for example, measure the amount of love each of them feels, and accordingly, decide whether to make them conscious of their feeling. Given his limited influence as an outsider, his behavior suits a finite state machine model (Figure 8). Research has shown that the harmonic oscillation may stabilize by discrete strategies containing two, three, of four states (Arstein, 1995). The relevance to the

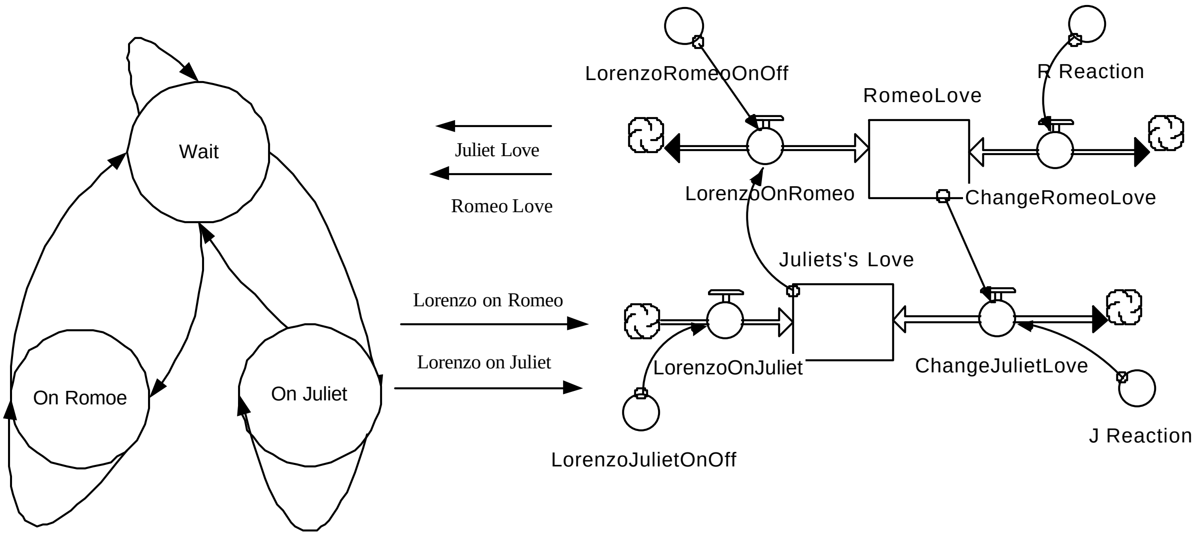


Figure 8 Romeo, Juliet and Lorenzo

consulting profession requires further discussion.

6. Summary

The paper described how hybrid modeling may be combined with the system dynamics methodology in order to serve didactical and communicative goals. To appreciate the full potential of this suggestion, further research and development are required. Our research seeks to clarify the cases suitable for hybrid modeling, and to evaluate their pedagogical value empirically.

Bibliography

1. Alessi, Stephen (2000). Designing educational support in system-dynamics-based interactive learning environments. *Simulation & Gaming* 31(2). 178-196
2. Artstein Zvi (1995) Examples of Stabilization with Hybrid Feedback in Rajeev Alur, Thomas A. Henzinger, Eduardo D. Sontag (Eds.): Hybrid Systems III: Verification and Control, Proceedings of the DIMACS/SYCON Workshop. p. 173-185
3. Asarin E., Bournez O., Dang T., Maler O., Pnueli A., (2000). Effective Synthesis of Switching Controllers for Linear Systems. *Proceedings of the IEEE* 88, No. 7. 1011-1025
4. Branicky, Michael (1995). *Studies in Hybrid Systems: Modeling, Analysis, and Control*. Unpublished doctoral dissertation. MIT, Cambridge.
5. Breedveld Peter C. (*1999)* On state-event constructs in physical system dynamics modeling, *Simulation Practice and Theory, Volume 7, Issues 5-6, 15, Pages 463-480*
6. Bunge, Mario (1974). *Treatise on Basic Philosophy. Volume 4: A World of Systems.* Dordrecht: Reidel Publishing.
7. Forrester, W. Jay (1961*) Industrial Dynamics*. Cambridge, MA: MIT Press.
8. Gupta, V. Jagadeesan R and Saraswat V. A (1998). Computing with continuous change, *Science of Computer Programming*, Volume 30, Issues 1-2, Pages 3-49
9. Hillis, W. Daniel. (1998). The pattern on the stone: the simple ideas that make computers work. New York: Basic Books
10. Labinaz Gino, . Bayoumi M Mohamed and Rudie Karen (1997). A survey of modeling and control of hybrid systems, *Annual Reviews in Control*, Volume 21, Pages 79-92
11. Lacoste-Julien, Simon (2002). MSDL Talk: Intro to Hybrid System Modelling. http://moncs.cs.mcgill.ca/people/slacoste/research/talk/MSDLtalk.html#tth_sE

c3.1

12. Levin I, Lieberman E. (2000) "Developing Analytical and Synthetic Thinking in Technology Education", Proceedings of International Conference on Technology Education, Braunshweig, Germany.
13. Levin, Tzur; Levin, Ilya (2002). *Hybrid Systems Modeling in Learning Science and Technology.* Journal of Computers in Mathematics and Science Teaching 21(4), 313-330
14. Maler, O. (2003) *Control from Computer* Science Annual Reviews in Control (to appear)
15. Maler, O. (2001) *Guest Editorial: Verification of Hybrid Systems European Journal of Control* 7, 357-365.
16. Mosterman Pieter J. & Biswas Gautam B. (2000); A comprehensive methodology for building hybrid models of physical systems. *Artificial Intelligence*, 121, 171–209.
17. Ossimitz G. (2002) *Stock-Flow-Thinking and Reading stock-flow-related Graphs: An Empirical Investigation in Dynamic Thinking Abilities*" Proceedings of the System Dynamics Conference, Palermo, Italy
18. Radzicki, Michael (1993). Dyadic Processes, Tempestuous Relationships, and System Dynamics. *System Dynamics Review* 9(1): 79-94.
19. Richardson, George (1986). Problems with Causal-Loop Diagrams, *System Dynamics Review,* 2,2 158-170.
20. Richardson, George (1991). Feedback Thought in Social Science and Systems Theory. Philadelphia: University of Pennsylvania Press.
21. Richardson, George (1997). Problems in Causal-Loop Diagrams Revisited. *System Dynamics Review* 13,3. 247-252.
22. Sterman J. (2000). *Business Dynamics: Systems Thinking and Modeling for a Complex World*. Irwin/McGraw-Hill: New York
23. Sterman, John D., Linda Booth Sweeney (2002). Cloudy skies: assessing public understanding of global warming. System Dynamics Review. Volume 18, Issue 2,. Pages: 207-240.
24. Strogatz, S. H. (1988). *Love affairs and differential equations*. Mathematics Magazine 61:35
25. Sweeney, Linda Booth & Sterman, John (2000). Bathtub dynamics: initial results of a systems thinking inventory. *System Dynamics Review,* 16(4), 249-286.