Reasoning About Energy in Qualitative Simulation

Pierre Fouché and Benjamin J. Kuipers

Abstract—Qualitative modeling and simulation make it feasible to predict the possible behaviors of a mechanism consistent with an incomplete state of knowledge. Though qualitative simulation predicts all possible behaviors of a system, it can also produce spurious behaviors, i.e., behaviors that correspond to no solution of any ordinary differential equation consistent with the qualitative model. A method for reasoning about energy is presented that eliminates an important source of spurious behaviors. This method is applied to an industrially significant mechanism—a nonlinear, proportional-integral controller—and show that qualitative simulation captures the main qualitative properties of such a system, such as stability and zero-offset control. It is believed that this is a significant step toward the application of qualitative simulation to model-based monitoring, diagnosis, and design of realistic mechanisms.

I. INTRODUCTION

QUALITATIVE modeling and simulation make it feasible to predict the possible behaviors of a mechanism consistent with an incomplete state of knowledge. Qualitative modeling can be useful for monitoring complex mechanisms, hard to model analytically or numerically, or for diagnosis of faulty mechanisms, whose model is unknown by definition.

An important aspect of qualitative simulation is that the behavioral prediction can branch, corresponding to qualitatively distinct futures that cannot be discriminated by the available information. Qualitative simulation is guaranteed to predict all real behaviors of systems consistent with the model. However, any given set of qualitative constraints may not be sufficiently powerful to filter out all inconsistent behaviors [11]. Such remaining behaviors, which correspond to no solution of any ordinary differential equation consistent with the qualitative model, are called spurious behaviors.

Spurious behaviors pose a significant efficiency problem for applications of qualitative simulation. When a model-based diagnostic engine matches observations against behaviors derived from a model, spurious behaviors consume time and effort, but more importantly may prevent a model from being eliminated once all its genuine behaviors fail to match. As we shall see, reasoning about energy makes it possible to eliminate an important source of spurious behaviors. In the first part of this paper, we will briefly review the central concepts of qualitative simulation, then use two simple examples to demonstrate

P. Fouché is with the Département Génie Informatique CNRS URA 817, Université de Compiègne, BP 639, 60206 Compiègne, France.

B. J. Kuipers is with the Department of Computer Sciences, the University of Texas at Austin, Austin TX 78712, USA.

sources of spurious behaviors, and their elimination using the energy constraint.

With these methods in hand, we turn our attention to an industrially significant problem: modeling and simulation of a controlled mechanism. A mechanism with its controller form an homeostatic system whose operation can conceal the manifestation of an underlying fault. In order to do model-based monitoring and diagnosis of such a system, we must be able to simulate correctly the behavior of a system including a realistic controller. Using the energy constraint, we obtain a tractable set of predictions for a nonlinear, proportional-integral controller, which allows us to deduce the qualitatively significant properties of such a mechanism, such as stability and zero-offset control. This suggests a new type of application for qualitative simulation, in the design of dynamic control systems, which will be discussed at more length in the conclusion.

II. QUALITATIVE SIMULATION OVERVIEW

Qualitative simulation is based on the observations that:
- the domain of a variable representing a physical parameter of a system can often be partitioned into a small number of "landmark" points and intervals between them, which represent real qualitative distinctions for the magnitude of the variable;
- knowing the direction of change of a variable, in conjunction with its qualitative magnitude, is often enough to determine the qualitative properties of its evolution; and
- for determining the qualitative behavior of a system, it is often adequate to know a functional relationship between two variables down to monotonicity and corresponding pairs of landmark values.

QSIM [10], [11], based on the above ideas, is a powerful simulation algorithm for deriving the possible qualitative behaviors of a system from its qualitative structural description. [14] provides a tutorial overview of qualitative simulation. [1] and [5] compare QSIM with the other qualitative simulation algorithms.

A qualitative model of a system consists of the variables that describe the system at a given level of abstraction, and the constraints that hold among the variables. Variables are continuously differentiable functions of time. The next step is to identify the landmarks, those values in the domain of the variable that make real distinctions for the behavioral description of the variable. For instance if one considers the amount of liquid in a tank then empty and full are such values. They are used to describe the qualitative magnitude (QMAG) of a variable, which is either a landmark, or an interval

Variables	Landmarks	Constraints	Corresponding Values
Y	0 +∞	d/dt(Y,V)	
V	-∞ 0 +∞	d/dt(V,A)	
A	-∞ g 0	constant(A)	

Fig. 1.   Simple ball model.

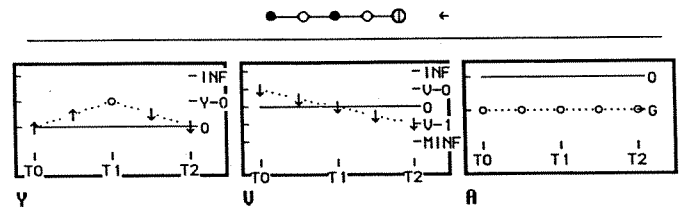| Variables	Landmarks	Constraints	Corresponding Values
Y	0 +∞	d/dt(Y,V)	
V	-∞ 0 +∞	d/dt(V,A)	
G	-∞ g 0	constant(G)	
F	-∞ 0 +∞	M^-(F,V)	$(-\infty,+\infty),(0,0),(+\infty/-\infty)$
A	-∞ g 0 +∞	add(G,F,A)	$(g,0,g)^2$

Fig. 2.   Ball with friction model.

Fig. 3.   Correct behavior of the simple ball.

Variables	Landmarks	Constraints	Corresponding Values
Y	0 +∞	d/dt(Y,V)	
V	-∞ 0 +∞	d/dt(V,A)	
-V-	0 +∞	abs-value(V,-V-)	
A	-∞ g 0	constant(A)	

Fig. 4.   Extended simple ball model.

between two adjacent landmarks. The totally ordered set of all the possible QMAGs of a variable is called its quantity space. A significant difference between QSIM and other algorithms is its ability to create new landmarks throughout simulation, in order to model new important qualitative distinctions. A variable is not only qualitatively described by its QMAG but also by its direction of change (QDIR), which can be either decreasing, steady or increasing. The pair (QMAG, QDIR) is called the qualitative value (QVAL) of a variable. The set of QVALs of all the system variables is the qualitative state of the system.

The set of all possible states of a system is restricted by relations between variables, called constraints. Constraints may model arithmetic relationships like $add(X, Y, Z)$, $mult(X, Y, Z)$ or $minus(X, Y)$, differential relationships like $deriv(X, Y)$ or functional relationships like $M^+(X, Y)$. The last constraint represents a very important class of functional relationships and states that some monotonically increasing function exists between X and Y. For instance, one can express the relation between the amount of liquid in a tank and its level by $M^+(LEVEL, AMOUNT)$. But one may also know that when the level is zero (resp. max) the tank is empty (resp. full), if these terms are landmarks in their respective quantity spaces. This knowledge is stored in the form of tuples of corresponding values associated with the constraint $M^+(LEVEL, AMOUNT) : (zero, empty)$ and $(max, full)$.

The set of all the variables with their associated quantity spaces and the set of all the constraints form a qualitative differential equation or a QDE. A system may evolve so that the current QDE may no longer be appropriate to describe its structure. For instance, the above M^+ relation is no longer valid if the tank overflows. The range of variables for which a QDE is valid is called an operating region. Functions describing how a system behaves when it goes out of some operating region are called region transitions.

Given a qualitative description of a system in terms of variables and constraints, QSIM starts from an initial state compatible with the QDE and determines the possible state(s) of the system during the interval of time immediately after the initial time-point. It then determines the possible changes of every variable that will lead to a qualitatively distinct state, defining a new time point t_1. This process is called limit analy-

sis. Possible changes are a variable reaching or moving from a landmark value or a variable changing its direction of change. If several possibilities are compatible with the constraints then QSIM will branch on every possibility. This potential for a branching sequence of events is an important difference between qualitative and numerical simulation. QSIM reapplies this process to every newly created state and the result is a tree of possible behaviors.

Qualitative simulation can produce total or attainable envisionments as well. Building a total envisionment consists in enumerating all the possible states and then determining the possible transitions between these states. An attainable envisionment is similar except that it starts from an initial state and incrementally builds the attainable states and transitions among them. A behavior is a path in the graph.

Taking into account only the direction of change of a variable is sometimes not sufficient to determine its evolution unambiguously, especially when its derivative is constrained only by continuity: when the variable becomes steady then it can be either increasing, decreasing or steady after that time and this produces an intractable set of behaviors. Introducing higher-order derivatives [2], [12], [15] allows the next direction of change to be determined unambiguously, since, in this case, it is given by the sign of the second derivative. Reasoning with higher-order derivatives turns out to be necessary to simulate complex, nonoscillatory systems. Kuipers and Chiu [12] also present a method for simulating a system without representing the direction of change of certain variables, when only their QMAGs are interesting. Fouché and Kuipers [7] discuss a comparative analysis of the currently available techniques in qualitative simulation.

III. SOME LIMITATIONS OF QUALITATIVE SIMULATION

We present two systems that are among the simplest for which the basic QSIM algorithm produces spurious predictions. By this we mean that some behaviors derived from a qualitative model do not correspond to any possible behavior of any system consistent with that model.

A. First Example: A Ball Thrown in the Air

Consider a ball thrown in the air, leaving the ground at time t_0 with an initial vertical speed V_0. If there is no friction only
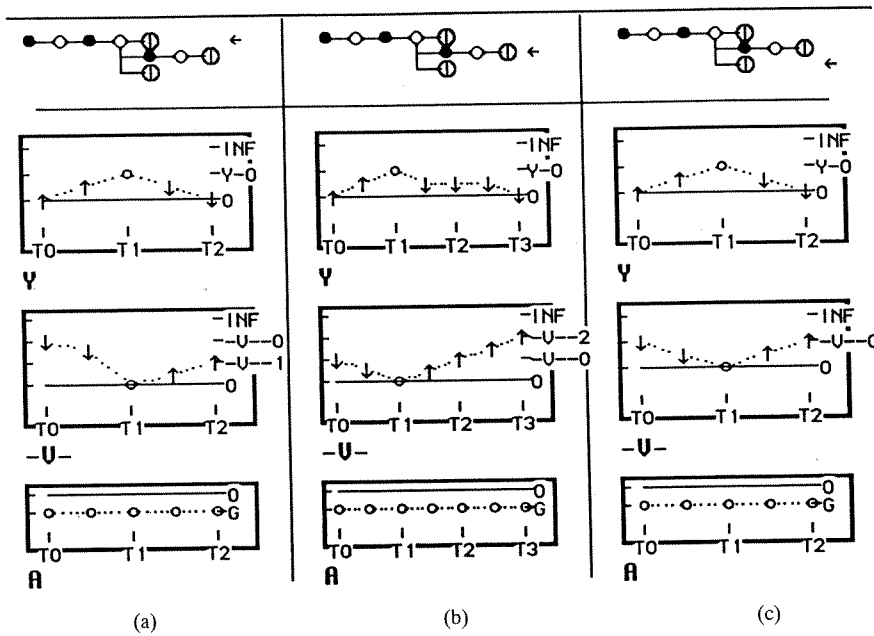
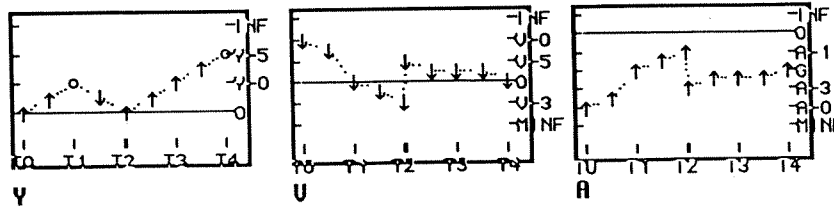Fig. 5.   Three behaviors of the extended simple ball.

Fig. 6.   Impossible behavior of the bouncing ball

one force is exerted on the ball and its acceleration is equal to constant gravity g. This yields the qualitative model of Fig. 1. We will refer to this system as the simple ball.

If we include friction in the model, the friction force F is inversely related to the speed of the ball. This is qualitatively modeled by $M^-(F, V)$. The fact that F is zero when the ball does not move is modeled by a pair of corresponding values (0,0) associated with the constraint. As the force would become infinitely large if the speed of the ball were infinite, we add the corresponding values $(-\infty, +\infty)$ and $(+\infty, -\infty)$. The qualitative model for this situation, referred to as the ball with friction, is given in Fig. 2.

Suppose first that there is no friction. The qualitative behavior of the ball is obvious: it first goes up for some period of time, then reaches a maximum height, creating a new qualitative time point t_1, then goes down before touching the ground at time t_2. This behavior is perfectly derived by QSIM, as shown in Fig. 3.

But what if we want to compare the value of the ball velocity at time t_0 and time t_2 ? Since there is no friction, the absolute value of the ball velocity should be the same. QSIM maintains a total ordering only between relative values and we have to introduce a new variable to represent the absolute value of the ball velocity, leading to the new qualitative model of Fig. 4. Unfortunately QSIM is unable to derive the correct behavior of the ball and branches on the three possible orderings when the ball touches the ground, as shown in Fig. 5.

This has a major consequence if we want to model a bouncing ball. Let us suppose that the shock is elastic (i.e., no energy is dissipated during the shock) and that the velocity of the ball is just reflected. Even if we do not represent the absolute velocity of the ball, QSIM will create three states after the bounce because it is unable to determine the ordering between the initial velocity and the velocity immediately after the bounce. Other spurious behaviors arise later in the simulation.

Suppose now that there is friction. In Fig. 6 the ball reaches a higher altitude after the first bounce with a lower initial velocity. Energy considerations allow us to explain why this is impossible: at time t_2, the energy of the ball is lower than at time t_0, but at time t_4, it is greater than at time t_1. As friction occurs the total mechanical energy of the ball should always be decreasing.

B. Second Example: The Spring-Block System

Let us consider another example where a similar problem occurs: a spring-block system (Fig. 7). It is one of the simplest mechanical system that exhibits oscillations, and it has been heavily used in the qualitative physics literature [3], [4], [8], [10], [11], [16]–[19]. It consists of a block connected to a spring laying on a horizontal table. The block position is referenced by a variable X, the origin being the rest position. Friction may or may not occur. The frictionless system will
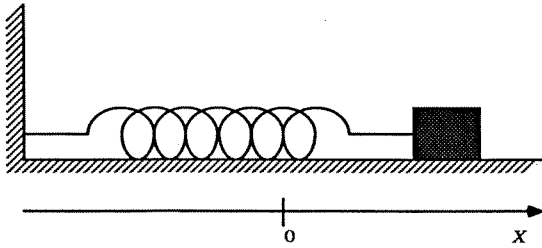
Fig. 7. The spring-block system.

Variables	Landmarks	Constraints	Corresponding Values
X	−∞ 0 +∞	d/dt(X,V)	
V	−∞ 0 +∞	d/dt(V,A)	
A	−∞ 0 +∞	M⁻(X,A)	(−∞,+∞),(0,0),(+∞,−∞)

Fig. 8. Model of the simple spring.

Variables	Landmarks	Constraints	Corresponding Values
X	−∞ 0 +∞	d/dt(X,V)	
V	−∞ 0 +∞	d/dt(V,A)	
F_S	−∞ 0 +∞	M⁻(F_S,X)	(−∞,+∞),(0,0),(+∞,−∞)
F_F	−∞ 0 +∞	M⁻(F_F,V)	(−∞,+∞),(0,0),(+∞,−∞)
A	−∞ 0 +∞	Add(F_S,F_F,A)	

Fig. 9. Model of the damped spring.

often be referred to as the simple spring, and the other as the damped spring. Though extremely simple from a structural point of view deriving their behaviors qualitatively turned out to be challenging.

We know that the force F_S exerted by the spring on the block is inversely related to its elongation. This is qualitatively modeled by $M^-(F_S, X)$. The fact that F_S is zero when the block is at its rest position is modeled by corresponding values (0,0) associated with the constraint. To model the fact that the force would become infinitely large if the block was moving toward infinity we add the corresponding values $(-\infty, +\infty)$ and $(+\infty, -\infty)$.

While the relation between F_S and X is frequently linear (Hook's law), we shall not make any linearity assumption, to demonstrate that our method applies to nonlinear systems. For the simple spring, no other forces are exerted on the block and we can directly model the acceleration as inversely related to the position of the block. For the damped spring, the friction force F_F is inversely related to the speed of the block and again we shall not assume that this relation is linear. The models are given in Fig. 8 and 9.

Simulating the simple spring produces one behavior until time t_3 and starts branching at time t_4. Fig. 10 shows the three possible behaviors at that time. In behavior (a), the block does not reach its initial position. In behavior (b), the block goes beyond and in behavior (c) the system comes back to its initial state.

Only the last behavior is genuine, of course. Furthermore, if simulation is allowed to continue after this point, stranger behaviors are produced: some of them correspond to decreasing oscillations followed by increasing oscillations, or vice-versa.

IV. POSSIBLE SOLUTIONS AND RELATED WORK

One way to eliminate these spurious behaviors is to introduce new variables representing potential and kinetic energy and to assert that the total energy is constant [10]. More generally, we can specify a Lyapunov function, generalizing the notion of energy, to represent the total energy of the system. This approach has several drawbacks:

- Finding such a function is generally very hard, especially for nonlinear systems. Additional variables must be added to the system model, reducing its simplicity. Here we are only interested in the position, speed and acceleration of the block.
- The simulator must be given additional information that is implicitly contained in existing constraints. It should be able to determine whether a system is conservative by analyzing the constraint structure.

Another approach is to perform envisionment as other qualitative simulation algorithms do [4, 3] and not create new landmarks (see Fig. 11), since in Fig. 10, branching occurs because new landmarks are created either for the position or the velocity.

This is adequate if one only wants to prove that the simple spring cannot reach quiescence. However:

- It is impossible to study the nature of oscillations because the position of the block at time t_4 cannot be compared with its initial position. Without creating new landmarks, the problem cannot even be expressed.
- Prohibiting the creation of new landmarks does not solve the problem, since an extended version of the simple spring including the variable $w = x - x_0$ exhibits the same spurious behaviors, branching on the ordering of $v = 0$ and $w = 0$.
- In an envisionment, just as in the basic QSIM algorithm, the validity of a transition depends only on the two adjacent states connected by the transition. Let us explain why the behavior in Fig. 10(a) cannot be ruled out by purely local reasoning: let S_1 be the state of the system during the interval $[t_3 t_4]$ and S_2 the state at time t_4 In S_1, there is no precise relationship between the speed and the position of the block, and so the transition between S_1 and S_2 is perfectly valid if we consider only S_1 and S_2. It is invalid in that behavior because we know that x_0 represents the initial position of the block, and at t_0, the block was dropped with no initial velocity. In another behavior, that transition might be valid. The validity of a transition depends on the behavior in which it takes place and global considerations must be introduced.

One way to reason globally about the behavior of a system is to use a phase space representation. The phase space for a system is the Cartesian product of a set of independent variables that fully describe the system. In practice, it provides another view of a system behavior: a system state is represented by a point in the phase space, and a behavior is represented by a trajectory. The phase space is a powerful tool to study properties of dynamical systems. A major theorem about the existence and uniqueness of the solution of an autonomous system of ordinary differential equations (i.e., one without
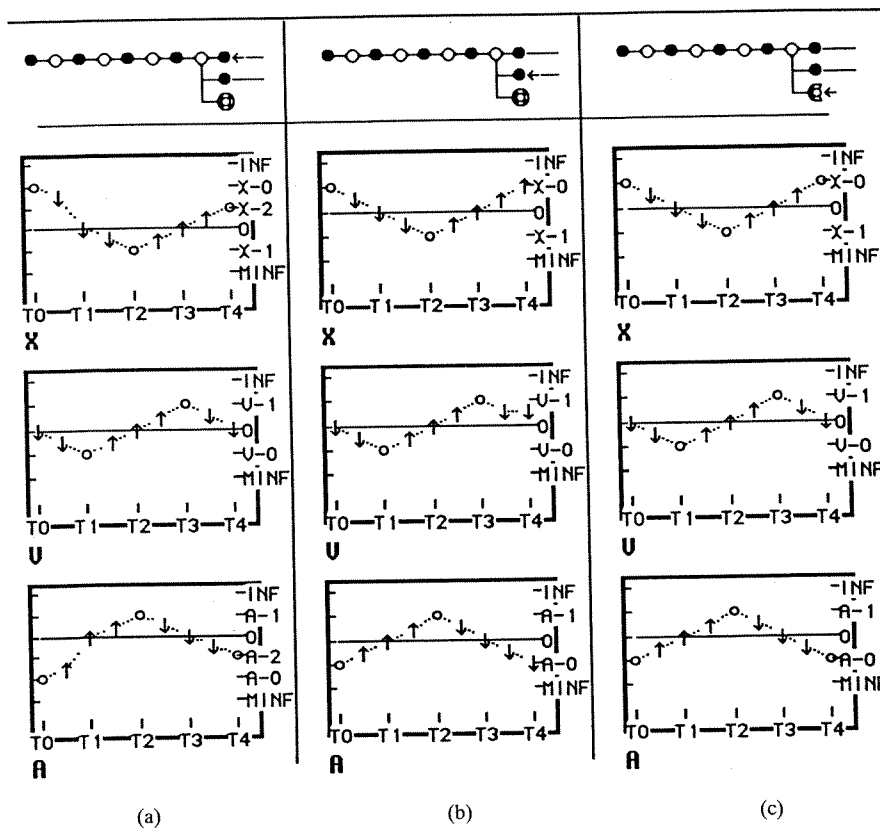
Fig. 10. Three way branching of the simple spring

time-varying input) has a direct equivalent in the phase space representation:

A trajectory that passes through at least one point that is not a critical point cannot cross itself unless it is a closed curve. In this case the trajectory corresponds to a periodic solution of the system.

Lee and Kuipers [16] and Struss [17] discovered independently that this property can be conveniently translated into qualitative terms for second order systems. In this case the phase space is a plane, so a general intersection criterion can be established even if trajectories are described qualitatively. This criterion can then be used to rule out trajectories that intersect themselves. One must note that in the quantitative case, only one among all possible pairs of independent variables is necessary to check the nonintersection property. But because reasoning qualitatively implies losing a certain amount of information, the nonintersection constraint must be applied to all possible phase planes: one qualitative phase plane may contain information that is not present in another one.

Struss [17] used the phase space representation in conjunction with symmetry properties of the simple spring to determine that its behavior was purely cyclic, but he could not use symmetry properties for the damped spring and both increasing or decreasing oscillations remained possible. However his technique proved that the nature of oscillations was a system property that could not change through time. In other words, "once damped, always damped." Lee and Kuipers' method [16] had the same limitations.

Ishida [8] used the qualitative theory of linear systems developed in econometrics. Introducing the notions of sign stability and sign observability he concluded that "oscillations will always converge eventually" for the damped spring and "the [simple spring] system is always in a pure periodical mode". However his method is restricted to linear systems, whereas these properties of oscillations in spring systems are independent of systems being linear or nonlinear.

V. ENERGY-BASED FILTERING

Reasoning about energy has an important place in our common sense understanding of the physical world. Many of the conclusions that one can draw about the evolution of a system are based on a qualitative analysis of the energy balance.

An explanation of the behavior of the mass-spring system to someone who is not familiar with mechanics might look like this:

- There is something that gives the system its ability to change its motion. Let us call this thing energy.
- Energy seems to be either contained in the block when it is moving or provided by the spring when it is stretched or compressed. Intuitively, the faster the block is moving the more energy it has; the more the spring is stretched (or compressed) the faster it will accelerate the block when the spring is released. Let us call the first, speed-related form kinetic energy and the second, which may potentially make the block move potential energy.
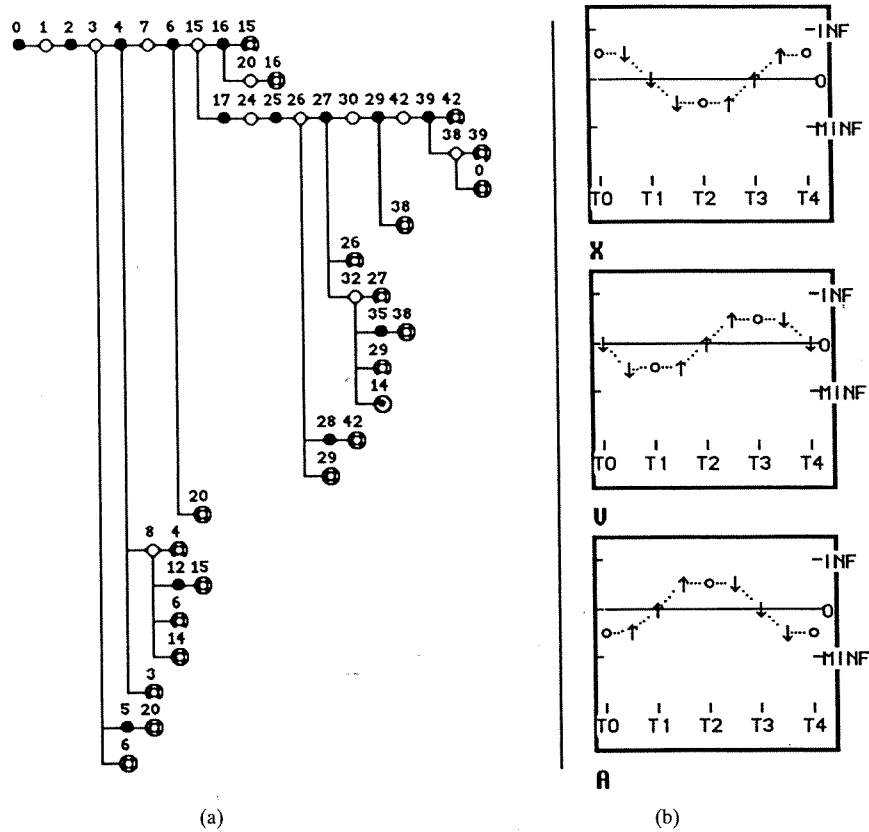
Fig. 11.   Attainable envisionments. (a) Envisionment graph of the damped spring. (b) Behavior of the simple spring.

• When the block is moving, for instance when it leaves the position where the spring is fully stretched, its potential energy is decreasing while its kinetic energy is increasing. Oscillations are a continual exchange between kinetic and potential energies.

• Friction causes energy to be lost in motion, so the block cannot reached the same extreme position each time the spring is stretched or compressed.

• When there is no friction, the sum of the potential and kinetic energies remains constant, so the block must reach the same extreme position when its speed is zero.

These explanations used qualitative descriptions of the concepts of mechanical energy, its conservation or dissipation, and two of its different forms: kinetic and potential. To generalize to nonmechanical systems, we have to formalize these intuitive notions. In classical mechanics, formalization is provided by both the Kinetic Energy Theorem and the decomposition of forces into conservative and nonconservative forces. However, these concepts are applicable to any second- or higher-order systems even when it does not make sense to talk about physical energy.

A. Principle

Theorem 1 (Kinetic Energy): Suppose we have a system including three variables X, V and A, continuous functions of time, such that:

$$\forall t \in [a, b], X'(t) = V(t) \text{ and } V'(t) = A(t) \qquad (1)$$

and $A(t)$ can be decomposed into two terms,

$$A(t) = C(X(t)) + N(t). \qquad (2)$$

Further, suppose the system moves, over the interval $[t_1 t_2]$ from

$$x_1 = X(t_1), v_1 = V(t_1) \text{ to } x_2 = X(t_2), v_2 = V(t_2)$$

then

$$\Delta K_e(t_1, t_2) = W_C(t_1, t_2) + W_N(t_1, t_2) \qquad (3)$$

where

$$\Delta K_e(t_1, t_2) = \frac{1}{2} m (v_2^2 - v_1^2) \qquad (4)$$

$$W_C(t_1, t_2) = m \int_{x_1}^{x_2} C(x) \, dx \qquad (5)$$

$$W_N(t_1, t_2) = m \int_{t_1}^{t_2} N(t) V(t) \, dt \qquad (6)$$

This theorem shows that the change in kinetic energy can be decomposed into the sum of the conservative work W_C and the nonconservative work W_N. The constant mass m appears here only for terminological compatibility with the physical terms "force," "energy," and "work." This result depends only on the equations relating X, V, A, C, and N and it applies equally well to nonphysical domains. Following the analogy with mechanical systems, we shall refer to X as the "position," V the "velocity" and A the "acceleration" of the system.

B. Qualitative Interpretation

In this paragraph, we will use the following notations to deal with sign of quantities: $[x]_a$ denotes the sign of $x - a$. When $a = 0$, we will simply use $[x]$. The sign of a continuously varying quantity f over an interval $[x_1 x_2]$ will be represented by $[f]_a^{[x_1 x_2]}$, and defined as follows:

- $[f]_a^{[x_1 x_2]} = 0$ iff $\forall x \in [x_1, x_2], f(x) = a$.
- $[f]_a^{[x_1 x_2]} = +$ iff $\forall x \in [x_1, x_2], f(x) \geq a$ and

$$\exists \, x_0 \in [x_1, x_2], f(x_0) > a.$$

- $[f]_a^{[x_1 x_2]} = -$ iff $\forall x \in [x_1, x_2], f(x) \leq a$ and

$$\exists \, x_0 \in [x_1, x_2], f(x_0) < a.$$

- $[f]_a^{[x_1 x_2]} = ?$ otherwise.

The key observation is that the sign of the quantities defined by (4), (5) and (6) can be computed using information present in a qualitative behavioral description, and that (3) can be checked using the classical sign addition law: The following theorems allow us to interpret qualitatively the kinetic energy theorem.

Theorem 2:

$$[\Delta K_e(t_1, t_2)] = [|v_2|]_{|v_1|} \tag{7}$$

The sign of the variation of kinetic energy depends only on the absolute value of V at t_1 and t_2. This sign can be evaluated if v_2 and v_1 have the same sign or if the absolute value of v is explicitly represented as a variable.

Theorem 3: if $[C(x)]^{[x_1 x_2]} \neq ?$, then

$$[W_C(t_1, t_2)] = [x_2]_{x_1} [C(x)]^{[x_1 x_2]}. \tag{8}$$

If $C(x)$ does not change sign on the interval between x_1 and x_2 then the sign of W_C depends only on that sign and on the ordering between x_1 and x_2:

Theorem 4: if C is odd over $[x1, x2]$ and if $[C(x)]^{[|x1||x2|]} \neq ?$, then

$$[W_C(t_1, t_2)] = [|x_2|]_{|x_1|} [C(x)]^{[|x_1||x_2|]}. \tag{9}$$

Even if $C(x)$ does change sign on the interval between x_1 and x_2, if we have more information about the nature of C (here, C is odd), it is still possible to compute $[W_C(t_1, t_2)]$ if $C(x)$ does not change sign on the shorter interval between $|x_1|$ and $|x_2|$.

Theorem 5: if $[N(t)V(t)]^{[t_1 t_2]} \neq ?$, then

$$[W_N(t_1, t_2)] = [N(t)V(t)]^{[t_1 t_2]}. \tag{10}$$

The sign of W_N can be determined only if the product $N(t)V(t)$ has constant sign from t_1 to t_2. Fortunately this is often true, since N_c frequently represents frictional or motor "forces", in the opposite or the same direction as the velocity.

The energy constraint is applied as a global filter. For each new state at a time point t_j, each earlier time-point t_i is considered. If the signs of the terms in (3) can be determined over the interval $[t_i t_j]$, and if they violate the qualitative addition constraint, the proposed state at t_j is inconsistent and can be filtered out.

C. Computational Complexity

It is easy to see that using the energy constraint requires $O(NL^2)$ computations of signs (signs of ΔK_e, W_C and W_N), where N is the total number of behaviors in the tree and L the average number of time points in all the behaviors. This sounds like a lot of additional work to do, but a very important point is that the energy constraint can early detect that some branches correspond to spurious behaviors and prevent those branches from being further developed. In practice this often results in computational time saving and much smaller behavior trees, compared to those obtained without the energy constraint (see Fig. 15).

D. Identifying the Energy Constraint

QSIM needs to be given the names of the variables X, V, C and N in order to apply the energy constraint. Algebraic expressions representing C and N in terms of other variables can be given as well, if these variables are not explicitly represented in the QDE. This can be done by hand, or by an algebraic manipulator that first looks for a derivative chain in the QDE, and then tries to identify an appropriate decomposition for the highest derivative in that chain.

The main problem in this search is to find a conservative term that is "as conservative as possible". It is always possible to specify that $N(t) = A(t)$ and $C(x) = 0$, but in that case the sign of the nonconservative work is almost always unknown. Our algebraic manipulator currently handles cases where the amount of manipulation is not too big, but fails to provide a useful answer for more complex systems such as the example of a PI-controller in Section VI, where the decomposition is not trivial.

This currently is the main limitation of our method. We are still improving our algebraic manipulator so it can handle more cases, but we are also investigating to see if other, more powerful manipulators, such as Macsyma or Mathematica, could be used instead, and interfaced with QSIM.

VI. RESULTS

We begin by explaining how the energy constraint works on the ball systems.

A. The Ball Systems

In our examples, the decomposition of the acceleration into conservative and nonconservative terms is trivial:

- For the simple ball: $C(x) = g$ and $N(t) = 0$.
- For the ball with friction: $C(x) = g$ and $N(t) = F(t)$.

Provided by the algebraic manipulator with the decomposition for the ball system, QSIM is able to determine that the two first behaviors of Fig. 5 are inconsistent, with the following justifications.

- Behavior a:
  "Inconsistent: between t0 and t2,
  Ke-var = -, C-work = 0, NC-work = 0"
  This means that between time points t_0 and t_2, the variation of kinetic energy is negative ($|V(t_0)| > |V(t_2)|$),
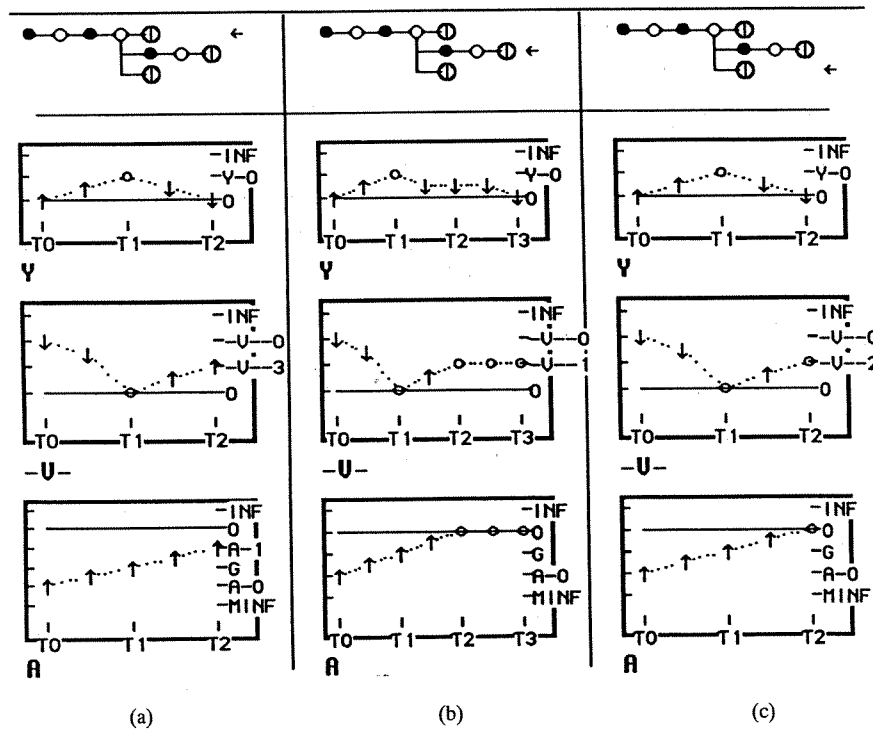
Fig. 12. Behaviors of the ball with friction

but both conservative and nonconservative works are zero $(Y(t_0) = Y(t_2)$ and $N_c(t) = 0)$

- Behavior b:
  "Inconsistent: between t0 and t2,
  Ke-var = 0, C-work = -, NC-work = 0"
  Here is the explanation:

  — $sign(\Delta K_e(t_0, t_2)) = 0$ because $|V(t_0)| = |V(t_2)|$
  — $sign(W_C(t_0, t_2)) = -$ because between t_0 and t_2, $g < 0$ and $Y(t_0) < Y(t_2)$
  — $sign(W_{N_c}(t_0, t_2)) = 0$ because $N_c(t) = 0$

Note that an inconsistency is detected before time t_3 and that simulation is stopped at time t_2.

The simulation of the ball with friction produces three behaviors shown in Fig. 12.

The important point to observe in Fig. 12 is that the absolute value of the velocity when the ball hits the ground is always lower than its initial value.

About Asymptotic Behaviors QSIM produces three behaviors because the ball may reach its terminal velocity (i.e., the velocity at which the friction force compensates exactly for the force of gravity) before or at the same time that the ball hits the ground. Strictly speaking the ball can never reach its terminal velocity. However one can argue that solutions b and c are intuitively appealing since in reality many falling objects quickly reach a speed that is indistinguishable from their terminal velocity. QSIM thus allows certain types of asymptotic motion toward a limit to be described as taking place in finite time: an example of technically spurious, but heuristically useful, predictions. Certain optional filtering methods are available [15], which are partially successful in filtering out these behaviors, if desired by the user.

Perhaps surprisingly, when we simulate a bouncing ball with elastic collisions but friction due to air resistance, we still get exactly three behaviors, all representing bounces of decreasing amplitude (Fig. 13). There is a single three-way branch according to whether the ball reaches terminal velocity on the first bounce. After the first bounce, the energy constraint ensures that the ball cannot reach its terminal velocity.

B. Springs

Again the algebraic manipulator provides QSIM with the decomposition of the acceleration into conservative and nonconservative terms:

- For the simple spring: $C(X(t)) = A(t)$ and $N(t) = 0$.
- For the damped spring: $C(X(t)) = F_S(t)$ and $N(t) = F_F(t)$.

Applying the energy constraint to the simple spring, QSIM determines that the two first behaviors of Fig. 10 are inconsistent, leaving a single cyclic behavior, and providing the following justifications.

- Behavior a:
  "Inconsistent: between t0 and t4,
  Ke-var = 0, C-work = +, NC-work = 0"
- Behavior b:
  "Inconsistent: between t0 and t4,
  Ke-var = +, C-work = 0, NC-work = 0"

Simulating the simple spring until time t_4 took the same time: 0.3 s with and without the energy constraint. It is interesting to insert another variable into the simple spring model to represent the absolute value of the position. This allows us to compare the extreme positions on each side of the rest position. QSIM now gives us three genuine behaviors,
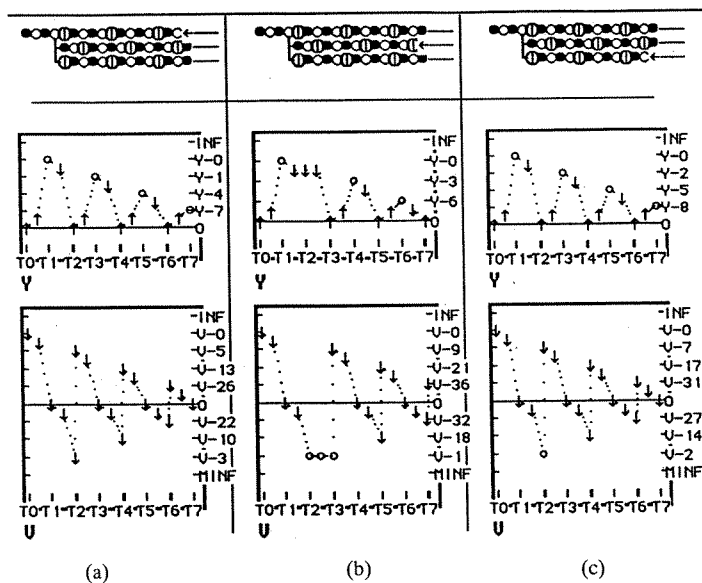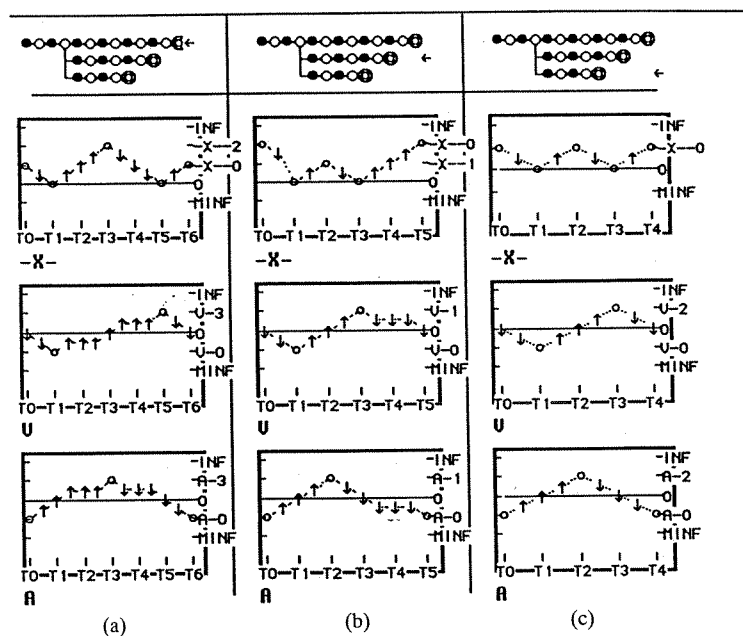
Fig. 13. Behaviors of the bouncing ball.

Fig. 14. Behaviors of the simple, nonsymmetric spring.

all cyclic. This branch happens because the spring force is not necessarily symmetric. The spring may be very stiff when compressed and weaker when stretched, or vice-versa, so the extreme positions may not be symmetric with respect to the rest position. Fig. 14 shows the three possible cyclic behaviors.

Applying the energy constraint to the damped spring allows QSIM to derive the correct behavioral description as shown in Figs. 15 and 16. We come up with one infinite, pseudocyclic behavior, exhibiting decreasing oscillations, and an infinite set of behaviors in which the system becomes overdamped after a finite number of oscillations. If the system were linear, it could not become over-damped after the first oscillation, but without a linearity assumption, these are genuine behaviors.

Simulating the damped spring until time t_9 took 4.3 s without the energy constraint and 1.6 s with it.

VII. A COMPLEX EXAMPLE: A NON-LINEAR PROPORTIONAL INTEGRAL CONTROLLER

A more complex example shows that, with the help of a variety of qualitative simulation methods, QSIM is able to derive the correct behaviors of a nontrivial system and to provide answers to interesting engineering questions.

The system in Fig. 17 is a tank whose level is controlled by a proportional-integral controller or PI-controller. This example is based on one proposed in [9]. Basically the principle of the controller is the following: a sensor provides the level l of liquid in the tank. The controller compares this value with a set point l_S and defines the difference between the two as the error e. The opening of the valve v is a function of the error and the integral i of error e. We have generalized the control law to include several monotonic function constraints, representing

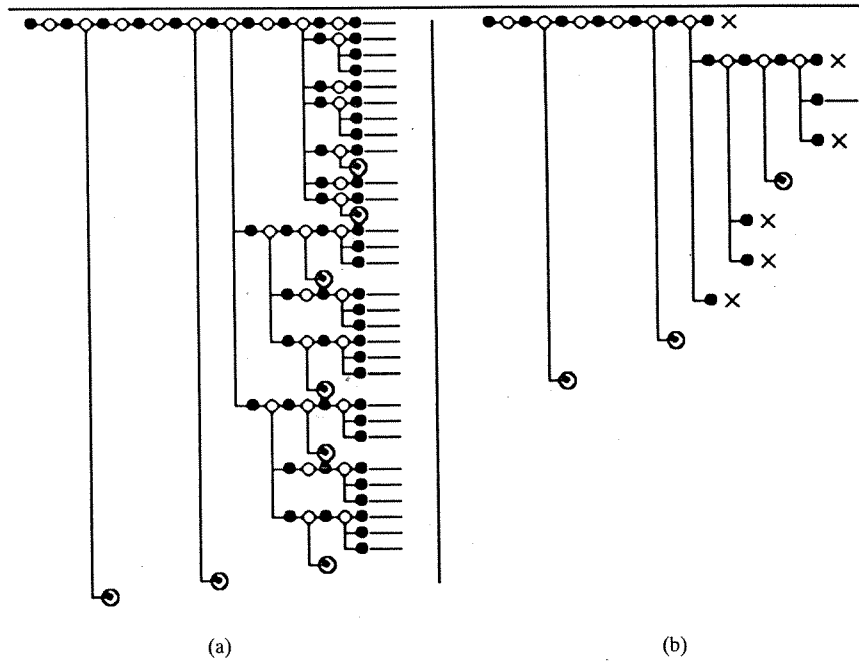(a)   (b)

Fig. 15.   Behavior trees for the damped spring. (a) Without the energy constraint. (b) With the energy constraint
(spurious behaviors that have been detected and labeled as inconsistent are indicated by an X).
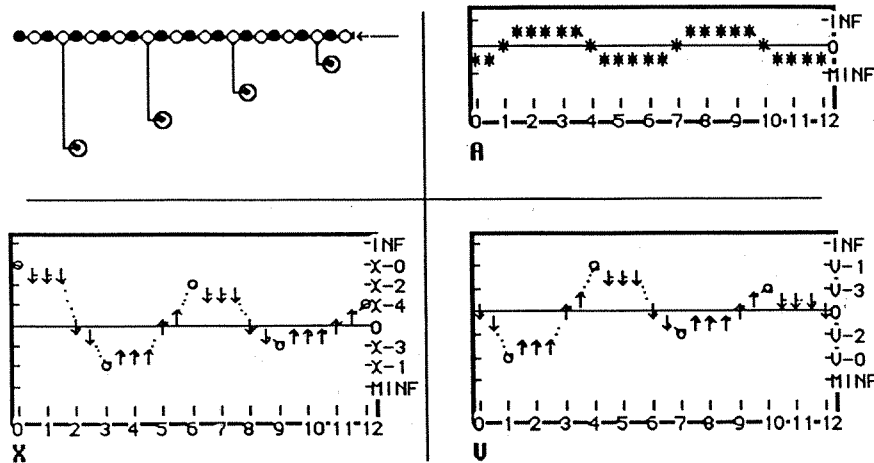

Fig. 16.   Decreasing oscillations of the damped spring.

unknown and potentially nonlinear functions. The conclusions we derive will therefore apply to any particular system that satisfies this qualitative description.

Modeling the system qualitatively is not straightforward. We outline the modeling process here; the derivation is given in detail in Appendix B. The system variables are presented in Fig. 18.

A. Model Building

The dynamic behavior of the system in open-loop is given (see Appendix B.1) by the equation

$$A\frac{dl}{dt} = q_1 - vC_v\sqrt{\rho a_g l}. \tag{11}$$

This equation is already nonlinear, but the main source of nonlinearity of the closed-loop system is the control law:

$$v = f_0(i) + g_0(e) \tag{12}$$
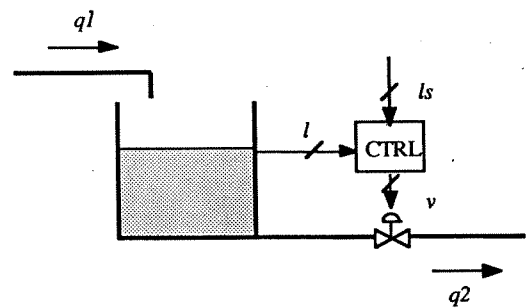

Fig. 17.   A level-controlled tank

where the only thing that we know about functions f_0 and g_0 is that

- they are instances of the M^+ class of functions: $(df_0(i))/(di) > 0$ and $(dg_0(e))/(de) > 0$
- $f_0(0) = g_0(0) = 0$.

q_1	volumetric inflow rate
A	cross-sectional area
P_a	ambient pressure
l	level of liquid in the tank
e	error
v	position of the valve
q_2	volumetric outflow rate
V_t	volume in the tank
P	pressure at the bottom of the tank
l_s	set point
i	integral of error
C_v	valve coefficient
q	volumetric netflow rate
ρ	liquid density
a_g	acceleration of gravity

Fig. 18. System variables

Initially the system is at equilibrium: $q = 0$, $e = 0$, $l = l_s$ and the inflow is at some value $q_1 = q_1^*$. We would like to study how this controller reacts to a step increase of the inflow: at time t_0, the inflow takes on the value q_1^+ with $q_1^+ > q_1^*$.

As we will compare the level to a fixed set point, we can introduce the following reduced variables:

$$F = \frac{1}{l_s}\frac{dl}{dt} \quad F_1 = \frac{q_1}{Al_s} \quad F_2 = \frac{q_2}{Al_s}$$

$$L = \frac{l}{l_s} \quad V = v \quad K = \frac{c_v}{A}\sqrt{\frac{\rho a_g}{l_s}}.$$

Substituting these variables into (11) yields the following equations:

$$L = E + 1 \quad F = \frac{dL}{dt} \tag{13}$$

$$F_2 = KV\sqrt{L} \quad F = F_1 - F_2$$

which can be directly transformed into QSIM constraints. Introducing auxiliary variables for the error and its integral:

$$E = \frac{e}{l_s} \quad I = \frac{i}{l_s} \tag{14}$$

allows the control equation (12) to be modeled qualitatively:

$$V = f(I) + g(E) \text{ with } f \in M^+ \text{ and } g \in M^+ \tag{15}$$

if functions f and g are defined by $f(I) = f_0(i) = f_0(l_s I)$ and $g(E) = g_0(e) = g_0(l_s E)$.

Without curvature and energy analysis the simulation is intractable. Let us analyze the system from the "energy" point of view first. The system composed of (13), (14), and (15) is a second-order system analogous to a spring: the "position" is the integral of error I, the "velocity" is the error E and the "acceleration" the netflow F. What we would like to have is a decomposition of F into two terms C and N, C depending only on I and N having a sign related to the sign of E, to be able to compute the sign of the nonconservative work. This decomposition is determined in Appendix B.2 and the result is:

$$C = F_1 - Kf(I) \tag{16}$$

$$N = -K(g(E) + Vh_1(E)) \tag{17}$$

with $h_1 \in M^+$ and $h_1(0) = 0$. F_1 and K are constant so the first term C is conservative and the sign of N is determined by the sign of E, for V is always positive.

B. Simulation Results

It is easy to see that the derivatives of variables V, F and F_2 are not constrained and that these variables will exhibit chatter. Ignoring their directions of change [12] eliminates this phenomenon. Fig. 19 shows the behavior trees computed until time t_5, using no global constraints, using the Non-Intersection Constraint (spurious behaviors are not deleted but labeled as inconsistent with a X at the end), using the Energy Constraint and using both of them (clockwise, from upper left corner). Used alone, the nonintersection constraint ensures that I and L must have similar behaviors (both increasing or both decreasing oscillations); the energy constraint ensures that I must exhibit decreasing oscillations.

Fig. 20 shows one particular behavior, with the energy and the nonintersection constraints, computed until time t_{12}.

1) All the time varying variables exhibit decreasing oscillations and the system can reach quiescence after an arbitrary number of oscillations. The controller is always stable in response to a step increase of inflow.
2) In every quiescent state the level is equal to the set point. There is no offset.

Since these properties are true in all the behaviors of the system, they are true for every physical system that is consistent with the qualitative model.

For the above analysis, we did not set an upper limit for the level in the tank. When we do provide such a limit, we get three behaviors at time t_1, two of which correspond to overflow or the level exactly reaching the limit without overflowing. This provides the additional conclusion that if the tank does not overflow during the first oscillation it will never overflow. Adding quantitative information to the model [13] may determine whether the tank will overflow or not, given numerical intervals for the step increase and other parameters of the system.

C. Using Curvature Constraints

We have not explicitly determined the behavior of the valve, since all that we know from Fig. 20 is that the valve is open. In order to determine its direction of change, a curvature analysis must be carried through [15]. This is explained in detail in Appendix B.3. Of course we have to apply the smoothness assumption to the functions f and g (i.e., the second derivatives of f and g are always small). If we also use the fact that the second derivatives of V and F are useful only when their first derivatives are zero, we obtain

$$V'' = F(f'(I) - KVg'(E)h_2'(E)) \tag{18}$$

$$F'' = -Kh_2(E)F\left(f'(I) + \frac{V'}{h_2^2(E)}\right) \tag{19}$$

with $h_2 \in M^+$ and $h_2(E) > 0$. The previous expressions are ambiguous if used directly. Under the assumption that the integral control term is "big enough" compared to $KVg'(E)h'(E)$ and $(V')/(h_2^2(E))$ then we have

$$\text{sign}(V'') = \text{sign}(F) \tag{20}$$

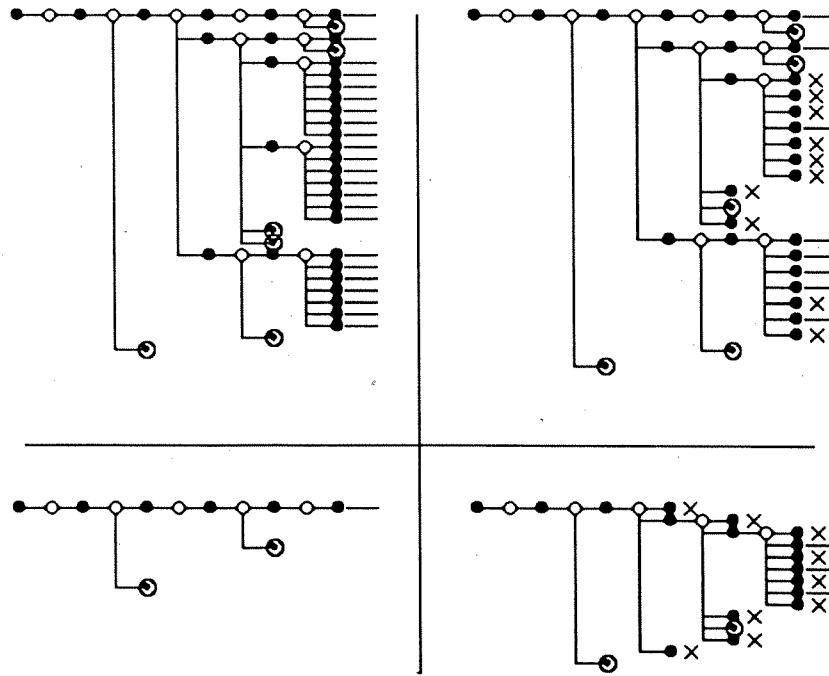$$\text{sign}(F'') = -\text{sign}(F). \tag{21}$$

Fig. 19.   Behavior trees of the level-controlled tank. Clockwise from upper left corner: without global constraints; with the nonintersection constraint; with the energy constraint; with both.
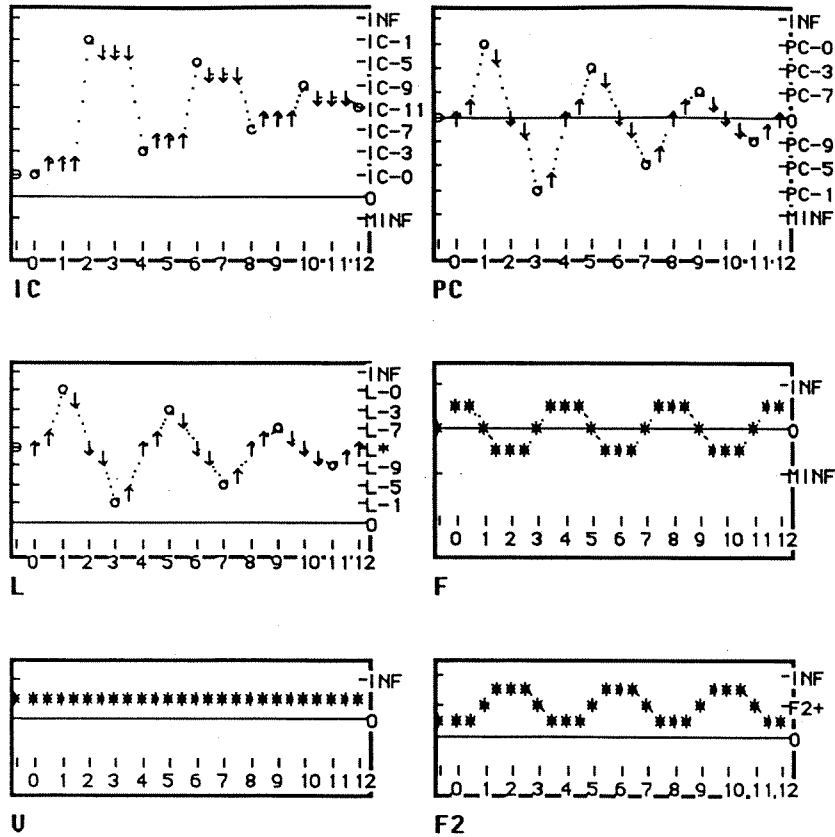

Fig. 20.   One behavior of the level-controlled tank This simulation provides answers to some questions a process control engineer would ask about the qualitative properties of the controlled system.

Fig. 21 shows three behaviors (only the valve, the level and the netflow are plotted). One remaining problem is that interpreting the assumptions we have made in terms of conditions on the parameters of the actual system is not trivial!

VIII. CONCLUSION

We have demonstrated the use of the energy constraint for eliminating an important class of spurious behaviors during qualitative simulation, and illustrated it with simple models
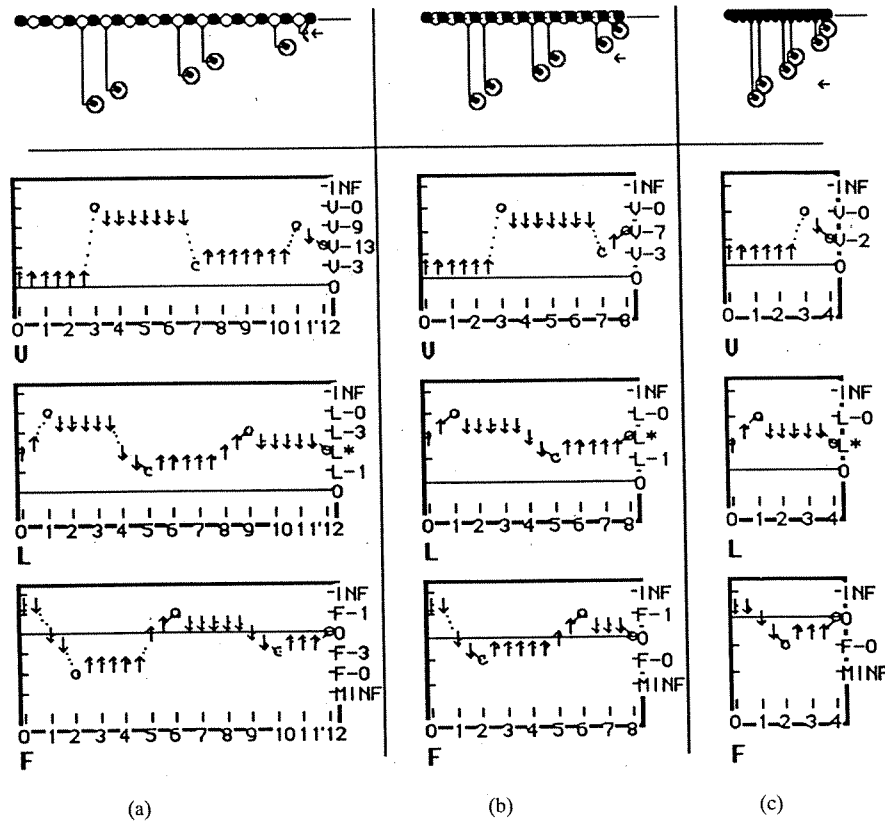
Fig. 21. Three behaviors of the level-controlled tank with simplifying assumptions.

of the ball and the spring, with and without friction. The more complex example of the PI controller demonstrates that the energy constraint can be applied, with some effort in the algebraic manipulation of the problem, to an industrially significant mechanism. This example also shows that qualitative simulation is now quite different from "naive physics." The goal of naive physics was to formalize our knowledge of the everyday physical world whereas qualitative simulation tends to provide a qualitative description of a system's possible behaviors, when that system is imperfectly known. It turns out that these subfields of AI, though very close at the beginning, evolved in a different way and the techniques used in qualitative simulation are anything but naive.

However, this example also suggests the value of qualitative simulation for the analysis of dynamic systems such as control systems. Qualitative simulation can be applied to mechanisms that are incompletely known, or which are completely characterized but include nonlinear elements, so that traditional analysis is ineffective. The soundness theorem for qualitative simulation [11], guarantees that all real behaviors of the mechanism will be among the set predicted by QSIM. If methods such as the energy and curvature constraints can produce a tractable set of predictions, then certain qualitative properties of the mechanism can be inferred by examining the properties of the qualitative behaviors. As we have seen with the PI controller, if properties such as stability or zero-offset are true of every predicted behavior, then they must be true of any mechanism satisfying the qualitative description.

The converse is also useful, though somewhat less so,

since qualitative simulation is not complete in its ability to filter out impossible behaviors. If an undesirable behavior appears among the set of predictions, it requires further careful analysis, to determine whether it is a genuine behavior of some mechanism satisfying the qualitative description, or whether the behavior is spurious. If it is genuine, it may be possible to impose additional qualitative or quantitative constraints on the system to prevent undesirable behaviors.

APPENDICES

A. Proofs

A1) The Kinetic Energy Theorem (Theorem 4.1): We can define the work done by the forces acting on the system from t_1 to t_2 as

$$W_T(t_1, t_2) = m \int_{t_1}^{t_2} A(t)V(t) \, dt \qquad (22)$$

Replacing A by V' in (22) and changing the variable of integration lets us write

$$W_T(t_1, t_2) = m \int_{t_1}^{t_2} V(t)V'(t) \, dt$$
$$= m \int_{v_1}^{v_2} v \, dv$$
$$= \frac{1}{2}m(v_2^2 - v_1^2).$$

By defining

$$\Delta K_e(t_1, t_2) = \frac{1}{2}m(v_2^2 - v_1^2) \qquad (23)$$

to be the change in kinetic energy in the system, we get

$$W_T(t_1, t_2) = \Delta K_e(t_1, t_2). \qquad (24)$$

Now, suppose we can decompose $A(t)$ into two terms, one of which depends only on the value of x:

$$A(t) = C(X(t)) + N(t). \qquad (25)$$

Then we can rewrite (1) as

$$W_T(t_1, t_2) = W_C(t_1, t_2) + W_N(t_1, t_2) \qquad (26)$$

where

$$W_C(t_1, t_2) = \int_{t_1}^{t_2} C(X(t))V(t)\,dt \qquad (27)$$

and

$$W_N(t_1, t_2) = \int_{t_1}^{t_2} N(t)V(t)\,dt. \qquad (28)$$

Substituting X' for V in (27) and changing the variable of integration gives us

$$W_C(t_1, t_2) = \int_{x_1}^{x_2} C(x)\,dx. \qquad (29)$$

Combining (23), (24), (26), (28), and (29) gives us the theorem.

A.2–Proof of Theorem 4.2:
- If $v_1 = v_2 = 0$, the identity is trivial.
- If $v_1 \neq 0$ or $v_2 \neq 0$, then $v_2^2 - v_1^2 = |v_2|^2 - |v_1|^2 = (|v_2| - |v_1|)(|v_2| + |v_1|)$.

Theorem 4.2 follows, for $[|v_2| + |v_1|] = +$

A.3–Proof of Theorem 4.3:
- If $t_1 = t_2$, then $x_1 = x_2$ and the identity is trivial.
- If $t_1 \neq t_2$, let us discuss according to the value of $[C(x)]^{[x_1 x_2]}$:

  + If $[C(x)]^{[x_1 x_2]} = 0$, the identity is trivial.
  + If $[C(x)]^{[x_1 x_2]} = +$, let us first consider the case $x_1 \leq x_2$. Since C is continuous and $C(x_0) > 0$,

$$\exists\, (x_0^-, x_0^+) \in [x_1 x_2]^2 : x_0^- < x_0 < x_0^+ \text{ and}$$
$$\forall\, x \in [x_0^- x_0^+], C(x) > 0.$$

  Hence, $\int_{x_0^-}^{x_0^+} C(x)\,dx > 0$. We can now decompose the integral

$$W_C(t_1, t_2) = \int_{x_1}^{x_0^-} C(x)\,dx + \int_{x_0^-}^{x_0^+} C(x)\,dx$$
$$+ \int_{x_0^+}^{x_2} C(x)\,dx.$$

  As $C(x) \geq 0$ over $[x_1 x_2]$, we have $W_C(t_1, t_2) > 0$. If $x_1 \geq x_2$ then $W_C(t_1, t_2) = -\int_{x_2}^{x_1} C(x)\,dx$ and we get $[W_C(t_1, t_2)] = [x_2]_{x_1}[C(x)]^{[x_1 x_2]}$

  + The proof extends without difficulty to the case $[C(x)]^{[x_1 x_2]} = -$.

A.4—Using Symmetries (Proof of Theorem 4.4):
- If $x_1 > 0$ and $x_2 > 0$, the identity is the same as in Theorem 4.3
- If $x_1 < 0$ and $x_2 < 0$, then

$$[|x_2|]_{|x_1|}[C(x)]^{[|x_1||x_2|]} = [-x_2 + x_1][C(x)]^{[-x_1 - x_2]}$$
$$= -[x_2 - x_1][-C(x)]^{[x_1 x_2]}$$
$$= [x_2]_{x_1}[C(x)]^{[x_1 x_2]}.$$

  From Theorem 4.3, this is equal to $W_C(t_1, t_2)$.
- If x_1 and x_2 have opposite sign, we can decompose the conservative work into two terms:

$$W_C(t_1, t_2) = \int_{x_1}^{0} C(x)\,dx + \int_{0}^{x_2} C(x)\,dx.$$

Changing the variable of integration x in the first integral to $-x$ yields

$$W_C(t_1, t_2) = -\int_{-x_1}^{0} C(-x)\,dx + \int_{0}^{x_2} C(x)\,dx.$$

Thus, using the fact that C is odd:

$$W_C(t_1, t_2) = \int_{-x_1}^{0} C(x)\,dx + \int_{0}^{x_2} C(x)\,dx.$$

Putting the two integrals together gives us

$$W_C(t_1, t_2) = \int_{-x_1}^{x_2} C(x)\,dx.$$

We are now back to one of the two cases we studied above, since $-x_1$ and x_2 have the same sign. This establishes the theorem.

A.5—Proof of Theorem 4.5: The proof is identical to the proof of Theorem 4.3, if we replace x by t and $C(t)$ by $N(t)V(t)$.

B. The PI-Controller

B.1—Analytic Model: The mass balance of the system in Fig. 17 is:

$$\frac{d}{dt}(\rho V_t) = \rho q_1 - \rho q_2. \qquad (30)$$

Given that the tank is cylindrical (of cross-sectional area A) and the density is constant, we get:

$$A\frac{dl}{dt} = q_1 - q_2. \qquad (31)$$

A valve is classically modeled by

$$q_2 = vC_v\sqrt{P - P_a} \qquad (32)$$

where v is the opening of the valve, C_v a coefficient characteristic of the valve and P_a is the atmospheric pressure. The pressure P at the bottom of the tank is

$$P = P_a + \rho a_g l. \qquad (33)$$

Putting together (31), (32), and (33) yields

$$A\frac{dl}{dt} = q_1 - vC_v\sqrt{\rho a_g l}. \qquad (34)$$

```
;;;----------------------------------------
Model of the PI-controller
;;;----------------------------------------
(Define-QDE PIC
 (text "A level-controlled tank")
 (quantity-spaces
  ;; Main variables
  (F1 (0 f1* f1+ inf))                        ;inflow
  (F2 (0 f2+ inf))                            ;outflow
  (F (minf 0 inf))                            ;netflow
  (L (0 l* inf)                               ;level in the tank
  (rL (0 rL* inf))                            ;square root of L
  (V (0 inf))                                 ;opening of the valve
 ;; Control variables
  (LS (0 l* inf))                             ;level set point
  (E (minf 0 inf))                            ;error: L-LS
  (PC (minf 0 inf))                           ;proportional control term
  (IC (minf 0 inf))                           ;integral control term
  (C (minf inf)))                             ;conservative term
 (constraints
  ;; System equations
  ((add F F2 F1) (0 f2+ f1+))                 ;mass balance
  ((d/dt L F))
  ((M+ rl L) (0 0) (rl* l*) (inf inf))
  ((mult V rL F2) )                           ;valve equation
  ;; Control equations
  ((add E LS L) (0 l* l*))                    ;error definition
  ((d/dt IC E))
  ((add PC IC V) ) ;PI control
  ((M+ PC E) (minf minf) (0 0) (inf inf))
  ((M- C IC) (minf inf) (inf minf))
  ((constant F1))
  ((constant LS)))
 (energy-constraint (IC E C (- E)))
  (ignore-QDIRs V F F2)
  (unreachable-values (v 0))
  (phase-planes (f2 c)(e c)(v e))
  (print-names (PC "g(E)")
               (IC "f(I)")))
;;;---------------------------
;;;; STEP-INCREASE simulates the model:
;;;; - steady-state is the equilibrium state,
;;;; - perturbed-state the state immediately after the step increase.
;;;---------------------------
(defun step-increase ()
 (qsim-cleanup)
 (let* ((*time-limit* 't12)
        (steady-state (make-new-state :from-qde PIC
p:assert-values '((E (0 std))
(f1 (f1* std))
(ls (l* std)))))
        (perturbed-state (create-transition-state :from-state steady-state
                                                  :to-qde PIC
                                                  :assert '((f1 (f1+ std))
                                                            (e (nil inc)))
                                                  :inherit-QMAG '(l ic ls))))
  (qsim-display (qsim perturbed-state)))))
```

Fig. 22. QMIS model.

The error e is defined by

$$e = l - l_s \qquad (6)$$

and its integral i by

$$i = \int_{-\infty}^{t} e(u) \, du. \qquad (36)$$

B. Energy Analysis

Consider the model defined by (13), (14), and (15). Let us begin by expressing $\sqrt{L}$ as a function of E:

$$\sqrt{L} = \sqrt{E+1}. \qquad (37)$$

To get the decomposition we must separate constant terms and terms that depend on E. Introduce a new function h_1 such that

$$\sqrt{E+1} = 1 + h_1(E). \qquad (38)$$

or equivalently

$$h_1(E) = \sqrt{E+1} - 1 \qquad (39)$$

where h_1 is an instance of M^+. Since $h_1(0) = 0$, $h_1(E(t))$ and $E(t)$ always have the same sign. Replacing $\sqrt{L}$ by its new expression in (13) yields

$$F_2 = KV + KVh_1(E). \qquad (40)$$

Hence

$$F = F_1 - K((f(I) + g(E)) + Vh_1(E)) \qquad (41)$$

or equivalently:

$$F = (F_1 - Kf(I)) - K(g(E) + Vh_1(E)) \qquad (42)$$

where F_1 and K are constant so the first term between parentheses is conservative. The important point concerning the second term is that its sign is determined by the sign of E, since V is positive. The appropriate decomposition is:

$$C = F_1 - Kf(I) \qquad (43)$$

$$N_C = -K(g(E) + Vh_1(E)). \qquad (44)$$

B.3–Curvature Analysis: Differentiating (15) yields

$$V' = Ef'(I) + Fg'(E) \qquad (45)$$

and

$$V'' = Ff'(I) + E^2 f''(I) + F'g'(E) + F^2 g''(E). \qquad (46)$$

In order to use (2) we shall assume that the functions f and g are relatively smooth so that their second derivatives are always very small (in the linear case they are equal to zero):

$$V'' = Ff'(I) + F'g'(E). \qquad (47)$$

Let us replace $\sqrt{L}$ in the expression of F_2 (13) by $h_2(E) = \sqrt{E+1}$

$$F_2 = KVh_2(E). \qquad (48)$$

Since F_1 is constant

$$F' = -F_2'. \qquad (49)$$

Deriving (48) yields

$$F' = -KV'h_2(E) - KVFh_2'(E) \qquad (50)$$

and

$$F'' = -KV''h_2(E) - 2KV'Fh_2'(E) - KVF'h_2'(E) \\ - KVF^2 h_2''(E). \qquad (51)$$

Using the smoothness assumption on h_2 yields

$$F'' = -KV''h_2(E) - 2KV'Fh_2'(E) - KVF'h_2'(E). \qquad (52)$$

Equations (47) and (52) are not useful directly. If we consider that (47) is useful only when $F' = 0$ and (50) when $V' = 0$, they can be simplified. Using $F' = 0$ in (6) yields

$$V'h_2(E) = -VFh_2'(E). \qquad (53)$$

From the definition of h_2, we know that

$$2h_2'(E)h_2(E) = 1 \qquad (54)$$

and (53) is reduced to

$$2h_2^2(E)V' = -VF. \qquad (55)$$

Using (55) in (52) yields

$$F'' = -Kh_2(E)\left(V'' + \frac{V'F}{h_2^2(E)}\right). \qquad (56)$$

Again using $F' = 0$ in (47) allows (56) to be simplified:

$$F'' = -Kh_2(E)F\left(f'(I) + \frac{V'}{h_2^2(E)}\right). \qquad (57)$$

To simplify (47), let us use $V' = 0$ in (45)

$$F = -E\frac{f'(I)}{g'(E)}. \qquad (58)$$

Putting this expression into (50) yields

$$F' = -KVFh_2'(E). \qquad (50)$$

Using (59) in (47) yields the final expression for V''

$$V'' = F(f'(I) - KVh_2'(E)g'(E)). \qquad (60)$$

B.4—QSIM Model: The QSIM model is shown in Fig. 22.

REFERENCES

[1] T. Bylander, "A critique of qualitative simulation from a consolidation viewpoint," *IEEE Trans. Syst., Man, Cybern.,* vol. 18, pp. 252–263, 1988.
[2] J. De Kleer, and D. J. Bobrow, "Qualitative reasoning with higher order derivatives," in *Proc. AAAI-84,* 1984, pp. 86–91.
[3] J. De Kleer, and J. Brown, "A qualitative physics based on confluences," *Artificial Intell. J.,* vol. 24, pp. 7–83, 1984.
[4] K. D. Forbus, "Qualitative process theory," *Artificial Intell. J.,* vol. 24, pp. 85–168, 1984.
[5] P. Fouché, A. Charles, J. P. Barthes, and C. Mélin, "Un panorama de la physique qualitative," in *Proc. Ninth Int. Workshop on Expert Systems and their Applications,* Avignon, France, 1989.
[6] P. Fouché and B. J. Kuipers, "Qualitative simulation of feedback control systems," in *Proc. MIM-S2 90,* Bruxelles, Belgium, Sept. 1990.
[7] P. Fouché and B. J. Kuipers, "An assessment of current qualitative simulation techniques," in *Recent Advances in Qualitative Physics,* P. Struss, and B. Faltings, Eds. Cambridge, MA: MIT Press, 1991.
[8] Y. Ishida, "Using global properties for qualitative reasoning: A qualitative system theory," in *Proc. IJCAI 89,* Detroit, MI, 1989, pp. 1174–1179.
[9] M. A. Kramer, "Expert systems for process fault diagnosis: A general framework," in *Computer Aided Systems for Process Operations,* G. V. Reklaitis, and H. D. Springgs, Eds. Amsterdam: Elsevier, pp. 557–587, 1987.
[10] B. J. Kuipers, "The limits of qualitative simulation," in *Proc. IJCAI-85,* Los Angeles, CA , 1985, pp. 128–136.
[11] B. J. Kuipers, "Qualitative simulation," *Artificial Intell. J.,* vol. 29, pp. 289–388, 1986.
[12] B. J. Kuipers, and C. Chiu, "Taming intractible branching in qualitative simulation," in *Proc. IJCAI-87,* Milan, Italy, 1987, pp. 1079–1085.
[13] B. J. Kuipers, and D. Berleant, "Using incomplete quantitative knowledge in qualitative reasoning," in *Proc. AAAI-88,* Saint Paul, MI, 1988, pp. 324–329.

[14] B. J. Kuipers, "Qualitative reasoning: Modeling and simulation with incomplete knowledge," *Automatica*, vol. 25, pp. 571–585, 1989.

[15] B. J. Kuipers, C. Chiu, D. Dalle Molle, and D. Throop, "Higher order derivative constraints in qualitative simulation," *Artificial Intell. J.*, vol. 51, pp. 343–379, 1991.

[16] W. W. Lee, and B. J. Kuipers, "Non-intersection of trajectories in qualitative phase space: A global constraint for qualitative simulation," in *Proc. AAAI-88*, Saint Paul, MI, 1988, pp. 286–290.

[17] P. Struss, "Global filters for qualitative behaviors," in *Proc. AAAI-88*, St. Paul, MI, 1988, pp. 275–279.

[18] L. Travé, and J. L. Dormoy, "Qualitative calculus and applications," in *Proc. 12th World Congress on Scientific Computation*, Paris, 1988.

[19] D. S. Weld, "Comparative analysis," *Artificial Intell. J.*, vol. 36, pp. 333–374, 1988.

Benjamin J. Kuipers (M'89) received the B.A. degree in mathematics from Swarthmore College, Swarthmore, PA, in 1970, and the Ph.D. degree in mathematics from the Massachusetts Institute of Technology, Cambridge, in 1977.

He is currently an Associate Professor of Computer Sciences at the University of Texas, Austin. He has worked at Harvard University, Tufts University, and the Massachusetts Institute of Technology. His research interests are in various aspects of common sense knowledge: qualitative reasoning with incomplete knowledge of physical mechanisms; resource-limited inference; and spatial exploration, learning, and problem solving.

Dr. Kuipers is a member of the American Association for the Advancement of Science, The Association for Computing Machinery, the Society for Values in Higher Education, and Computer Professionals for Social Responsibility.
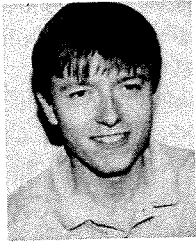
Pierre Fouché graduated from the Ecole Centrale de Paris in 1987. He currently is a Ph.D student in computer science at the University of Compiègne, France in the C.N.R.S.U.R.A. 817 laboratory.

He visited the Artificial Intelligence Laboratory at the University of Texas at Austin in 1989–1990. His research interests include qualitative reasoning about physical systems, planning of continuous, dynamic systems and applications of AI techniques to process control and supervision.