Higher-Order Derivative Constraints in Qualitative Simulation*

Benjamin Kuipers†
Charles Chiu‡
David T. Dalle Molle§
David Throop¶

1991

Artificial Intelligence **51**: 343-379, 1991
(with small changes to figures 8, 9, and 10).

*This work has taken place in the Qualitative Reasoning Group at the Artificial Intelligence Laboratory, The University of Texas at Austin. Research of the Qualitative Reasoning Group is supported in part by NSF grants IRI-8905494 and IRI-8904454, by NASA grant NAG 2-507, and by the Texas Advanced Research Program under grant no. 003658-175.

†Dept. of Computer Sciences, University of Texas at Austin, Austin, Texas 78712 USA
‡Dept. of Physics, University of Texas at Austin, Austin, Texas 78712 USA
§Amoco Chemical Co., Naperville, Illinois 60566 USA
¶Dept. of Computer Sciences, University of Texas at Austin, Austin, Texas 78712 USA

Abstract

Qualitative simulation is a useful method for predicting the possible qualitatively distinct behaviors of an incompletely known mechanism described by a system of qualitative differential equations (QDEs). Under some circumstances, sparse information about the derivatives of variables can lead to intractable branching (or "chatter") representing uninteresting or even spurious distinctions among qualitative behaviors. The problem of chatter stands in the way of real applications such as qualitative simulation of models in the design or diagnosis of engineered systems.

One solution to this problem is to exploit information about higher-order derivatives of the variables. We demonstrate automatic methods for identification of chattering variables, algebraic derivation of expressions for second-order derivatives, and evaluation and application of the sign of second- and third-order derivatives of variables, resulting in tractable simulation of important qualitative models.

Caution is required, however, when deriving higher-order derivative (HOD) expressions from models including incompletely known monotonic function (M^+) constraints, whose derivatives beyond the sign of the slope are completely unspecified. We discuss the strengths and weaknesses of several methods for evaluating HOD expressions in this situation.

We also discuss a second approach to intractable branching, in which we change the level of description to collapse an infinite set of distinct behaviors into a few by ignoring certain distinctions.

These two approaches represent a trade-off between generality and power. Each application of these methods can take a position on this trade-off depending on its own critical needs.

1 Introduction

Qualitative simulation predicts the possible qualitatively distinct behaviors of an incompletely known mechanism described by one or more qualitative differential equations (QDEs). The creation and simulation of qualitative models plays a critical role in supporting model-based reasoning about physical mechanisms in the face of incomplete knowledge. In diagnosis, the possible behaviors of an incompletely known fault model can be matched against observations; in design, the possible behaviors of a partially specified mechanism can be compared with desirable and undesirable properties of the final design. In both diagnosis and design, the strength of the qualitative representation is that a finite description can capture a state of incomplete knowledge of structure and the set of all possible behaviors.

The structure of a mechanism is described by a QDE: a collection of continuous variables and algebraic and differential constraints among them. Such a constraint model may be derived from a component-connection description [Sussman and Stallman, 1975; de Kleer and Brown, 1984; Williams, 1984a], from a process-view description [Forbus, 1984], or be given as part of the problem-solver's model of the domain [Kuipers, 1984; Kuipers and Kassirer, 1984]. One advantage of qualitative reasoning methods is the ability to express and reason with incomplete knowledge of functional relationships, describing them qualitatively as monotonically increasing or decreasing, and passing through certain corresponding landmark values. For example, one may say that wind resistance increases monotonically with velocity, without needing to know or assume the exact relationship: $resistance = M^+(velocity)$.

QSIM is a representation for QDEs that has a precise relationship with differential equations, and an algorithm for qualitative simulation with an efficient implementation [Kuipers, 1986]. QSIM takes as input a QDE (or system of QDEs) and a description of its state at time t_0. At each point in time, the value of each variable in a QDE is described qualitatively: its magnitude in terms of ordinal relations with a discrete set of *landmark values*, and its derivative in terms of direction of change. The fundamental operation in qualitative simulation is *limit analysis*: when several variables are changing and moving toward landmark values, the constraints in the QDE are analyzed to determine which combinations of limits may be reached and hence which qualitative states may come next.

The possible behaviors of the system are predicted as a (possibly branching) tree of qualitative states. A behavior is a sequence of qualitative descriptions of states:

$$Behavior = [state(t_0), state(t_0, t_1), state(t_1), \ldots, state(t_n)].$$

QSIM predicts a set of possible behaviors, which is interpreted as a disjunction:

$$QSIM : QDE, state(t_0) \longrightarrow or(B_1, \ldots B_k).$$

That is, starting in $state(t_0)$, QSIM predicts that one of the behaviors $B_1, \ldots B_k$ will describe the actual behavior of the system. This inference can be shown to be sound (i.e. the disjunction will always include the real behavior), but incomplete (i.e. there may be impossible disjuncts that the algorithm cannot filter out) [Kuipers, 1986].

The success of diagnostic, design, and other applications of qualitative simulation rests on the ability to produce a tractably small set of predictions including all real possible behaviors of the mechanism. In some cases, simulation of a QDE produces a small set of behaviors, all representing real possibilities consistent with the available knowledge. However, in other cases, the result may be an intractably branching tree of predicted behaviors. A few real solutions may be obscured by a forest of non-solutions, or all solutions may be real, but not interestingly distinct.

This problem arises from the incomplete qualitative descriptions of variable values: an ordinal description of the magnitude with respect to landmark values, the sign of the first derivative, and no information about higher derivatives. With such sparse information, circumstances arise where certain variables "chatter:" their behavior is unconstrained except by continuity. The simulation must then branch on every possible number, magnitude, and timing of changes of the chattering variables, resulting in an intractably branching, and hence useless, set of predictions. Figure 1 shows one behavior in an intractably branching tree of predictions for a system of two cascaded tanks. The behaviors are distinguished only by the behavior of the variable $netflowB(t)$, representing the rate of change of the amount in the second tank.

The presence of an infinite family of uninteresting behaviors is particularly striking when the set of possible behaviors is represented as a tree, as QSIM does (fig. 1). The same problem arises, however, in the finite state-transition-graph (or "total envisionment") representation of qualitative behavior [de Kleer and Brown, 1984; Forbus, 1984; Williams, 1984]. Once one attempts to interpret the transition graph as predicting specific behaviors, loops in the graph give rise to infinite families of paths, and the same problem of "chatter" arises.

In this paper, we present two complementary solutions to the problem of chatter, discuss their limitations and trade-offs, and present examples of their application. The first solution exploits higher-order derivative information implicit in the QDE to *eliminate* certain predicted behaviors. Derivation of higher-order derivatives in the presence of incomplete knowledge of monotonic function constraints requires an additional "sign-equality" assumption, assuming that monotonic functions are smooth in a certain way. We discuss this assumption in detail, providing conditions that guarantee that it is satisfied, and demonstrating the effect on the prediction when it is violated. The second solution changes the granularity of the qualitative description, to *collapse* many behaviors into a few. Changing the level of description avoids the need for an additional assumption, but the predicted behavior contains less information.

This paper is an outgrowth of our previous work [Kuipers & Chiu, 1987; Chiu, 1988; Dalle Molle, 1989a; Throop, 1989]. The methods presented here eliminate chatter in many realistically complex mechanisms [Dalle Molle, 1989b; Fouché& Kuipers, 1990], and are a necessary tool for eliminating spurious predictions generally. However, these methods are fundamentally local, constraining branching at particular time-points. There are also non-local sources of spurious predictions, which require correspondingly non-local constraints such as energy and system property constraints [Lee, Chiu, & Kuipers, 1987; Fouché& Kuipers, 1990], the qualitative analysis of the phase plane of the solutions to a QDE [Lee & Kuipers, 1988; Struss, 1988], and decomposition of large-scale mechanisms into weakly-interacting components by time-scale abstraction [Kuipers, 1987b; Simon & Ando, 1961; Iwasaki & Bhandari, 1988]. Kuipers [1989] provides a tutorial overview of the current state of qualitative simulation research.

The higher-order derivative constraint and all examples discussed in this paper have been implemented as part of the QSIM program. All of the qualitative graphs in this paper are QSIM output. The QSIM kernel is implemented in pure CommonLisp, and is available to interested researchers by contacting the first author.

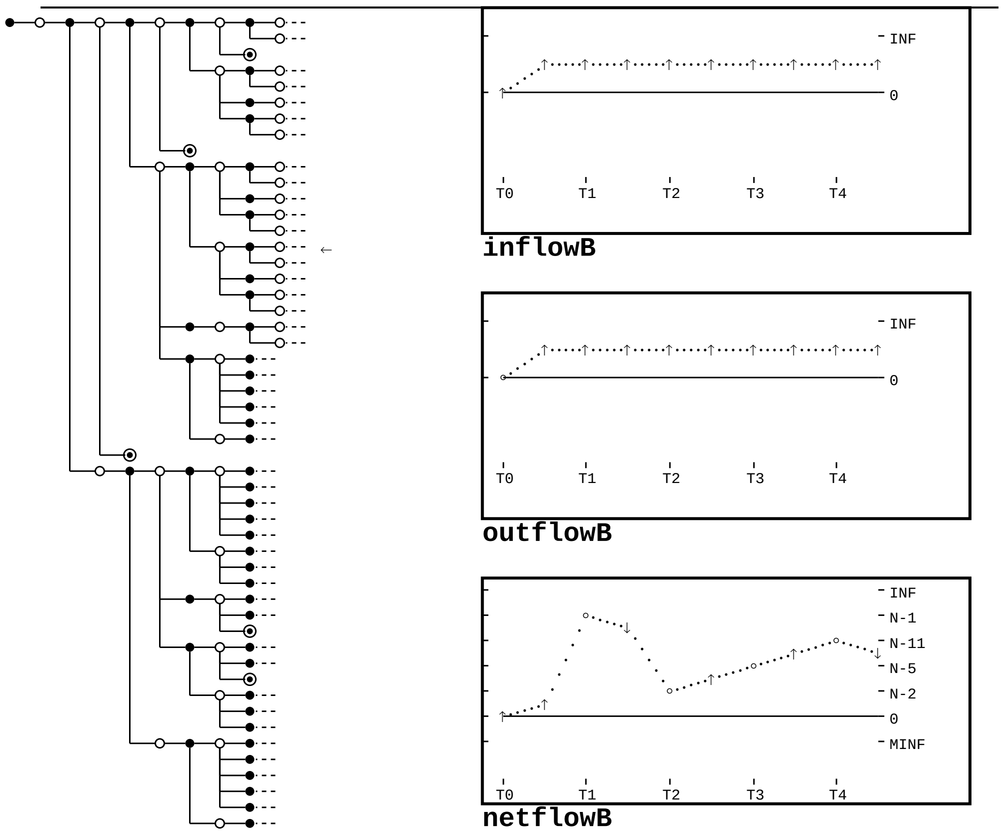
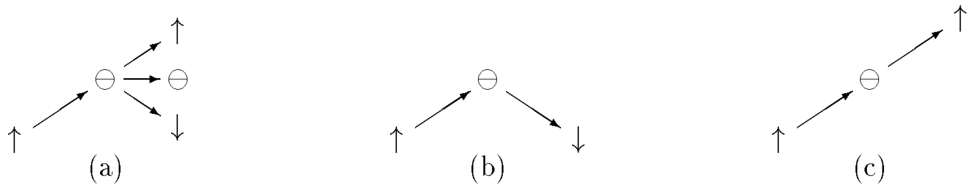


Figure 1: Intractable branching due to chatter
In a qualitative model of two cascaded tanks (A and B), $NetflowB(t) = InflowB(t) - OutflowB(t)$ is constrained only by continuity as long as it remains in the interval $(0, \infty)$. Thus, the simulation branches on all possible trajectories of $NetflowB(t)$, while all other variables have completely uniform behavior.



(a) $var(t_i)$ has a three-way branch from a critical point: $var'(t_i) = 0$. (The $inc \rightarrow std \rightarrow std$ behavior is only permitted under an option that allows non-analytic functions for $var(t)$. See Appendix A.)

(b) In case we know that $var''(t_i) < 0$, only one of three branches is consistent.

(c) If $var''(t_i) = var'(t_i) = 0$, and $var'''(t_i) > 0$, then only one branch is consistent.

Figure 2: Three-way and one-way branches

2 Higher-Order Derivatives

2.1 Introduction

The first method for eliminating chatter is based on knowledge of higher-order derivatives, implicit in the QDE, but neglected by the basic limit analysis methods [de Kleer and Brown, 1984; Forbus, 1984; Kuipers 1984, 1986].

Suppose that a variable $var(t)$ reaches a critical point: i.e. $var'(t_i) = 0$. Qualitatively, over the following qualitatively uniform interval (t_i, t_{i+1}), $var'(t)$ could be positive, negative, or zero (fig. 2a). In QSIM terminology, the direction of change $qdir(var)$ could be *inc*, *dec*, or *std* during the time-interval (t_i, t_{i+1}).

If the derivative of $var(t)$ is not adequately constrained, directly or indirectly, none of the three possibilities in fig. 2a can be excluded, so a branch is required. However (fig. 2b), if we have reason to know that $var''(t_i) < 0$, then two of these possibilities can be excluded, leading to a unique description of the qualitative state over (t_i, t_{i+1}).

More generally, at any time-point t_i, the sign of the first non-zero derivative of var at t_i determines the direction of change of var over (t_i, t_{i+1}).

Definition 1 *Just as $qdir(var)$ represents the sign of the first derivative of var, written $[var'(t)]$, we define the abbreviations $sd2$ and $sd3$ for the signs of the second and third derivatives of var.*

$$qdir(X,t) = \left[\frac{dX}{dt}(t)\right]; \quad sd2(X,t) = \left[\frac{d^2X}{dt^2}(t)\right]; \quad sd3(X,t) = \left[\frac{d^3X}{dt^3}(t)\right].$$

$sd1(var)$ may be used as a synonym for $qdir(var)$. The value *nil* represents an ambiguous sign. The second argument, t, to $qdir$, $sd2$, and $sd3$, may be suppressed when the current time-point is clearly specified by context.

We will use the term *higher-order derivative (HOD) constraint* to refer to the use of the first non-zero derivative to filter out impossible behaviors as in Fig. 2(b,c). In the usual case, it is the second derivative $var''(t_i)$ that provides the necessary information (fig. 2b), and we may then refer to the HOD constraint as a *curvature constraint.* In more complex situations (cf. fig. 2c, and section 3.2), third-order derivatives may be required. We do not extend our analysis beyond third-order derivatives, for reasons that will be discussed in section 4.

2.2 Building on Previous Work

Higher-order derivative information was first applied in qualitative simulation by Williams [1984a, 1984b] and by de Kleer and Bobrow [1984]. Williams [1984a, 1984b] showed that knowledge of the higher-order derivatives of an input — for example, that an input is not merely positive, but is a linearly increasing ramp — could be effectively propagated through constraints to reduce certain ambiguities. De Kleer and Bobrow [1984] showed how confluences representing higher-order derivatives could be derived from an ordinary differential equation for a mechanism and applied to reduce ambiguity in the prediction.

Our approach starts from these correct observations, and extends them in several ways. The most straight-forward approach to higher-order derivatives extends the constraint model to include terms for the higher-order derivatives and constraints linking them to the previous terms. Unfortunately, this simply pushes the problem of chatter into the higher-order terms, while adding variables whose distinctions may cause additional branching in the behavior tree.

Our methods are designed to apply to *qualitative* differential equations, representing incomplete knowledge of the structure of a mechanism. We discuss the relationship between the strength of conclusion one can draw, and the amount of knowledge one has in the QDE. We present methods that can, in many cases, determine when higher-order derivative reasoning is required, and automatically derive the appropriate constraints.

The algebraic methods we use to derive expressions for higher-order derivatives from the QDE are similar in spirit to those used in MINIMA [Williams, 1988]. The shared insight is that qualitative constraints are abstractions of constraints on real-valued variables, and so can be manipulated by traditional algebraic methods before being mapped into qualitative value spaces. After mapping an expression into a qualitative value space, further simplification is possible. Our implementation could be made more powerful by replacing its simple algebraic manipulator with MINIMA, at the cost of incorporating Macsyma [Macsyma, 1988] as a subsystem of QSIM.

There are three steps to applying the higher-order derivative constraint:

1. Identify variables in the QDE likely to chatter.
2. Derive algebraic expressions and evaluate them to obtain the signs of the second- or third-order derivatives of chattering variables.
3. Use the sign of the higher-order derivative to constrain branching.

We will discuss steps 1 and 3 before step 2, which raises more complex issues. Section 3 provides detailed examples of each step of the higher-order derivative (HOD) constraint, as it is applied to two- and three-tank cascade systems.

2.3 Identifying Chattering Variables

Definition 2 *A variable v appearing in a QDE* chatters*, starting at a qualitative time-point t_i, if the constraints in the QDE are consistent with any qualitative value of $qdir(v,t)$, for every t in some open interval (t_i, t_{i+1}).*

It is possible to propose candidate variables that are likely to chatter during simulation by analysis of the structure of the QDE. We observe, first, that if two variables x and y are related by a monotonic function constraint, either both chatter, or neither does. Second, if the derivative x' of a variable x is explicitly represented in the QDE with sufficient constraints, then the variable x will not chatter.[1]

[1] Suppose we have the pathological situation that both x and its derivative x' appear explicitly in a QDE, but both variables are otherwise unconstrained. According to the definition above, x' will chatter, while x will not. Although x is unconstrained, so QSIM will eventually predict all possible qualitative behaviors for x, $qdir(x)$ is always constrained by the sign of x'.

The algorithm for proposing candidate variables is as follows:

1. Group the variables in the QDE into equivalence classes according the following criteria:

$$\begin{aligned} equiv(x,y) &\leftarrow M^+(x,y) \\ equiv(x,y) &\leftarrow M^-(x,y) \end{aligned}$$

   We may exploit the fact that other explicit constraints in the QDE imply the weaker M^+ or M^- constraints [Kuipers, 1984, Appendix D]. For example,

$$\begin{aligned} equiv(x,y) &\leftarrow MINUS(x,y) \\ equiv(x,z) &\leftarrow ADD(x,y,z) \text{ and } constant(y) \\ equiv(x,y) &\leftarrow ADD(x,y,z) \text{ and } constant(z) \\ equiv(x,z) &\leftarrow MULT(x,y,z) \text{ and } constant(y) \\ equiv(x,y) &\leftarrow MULT(x,y,z) \text{ and } constant(z) \\ equiv(w,z) &\leftarrow ADD(x,y,z) \text{ and } M^+(w,x) \text{ and } M^+(w,y) \\ equiv(w,x) &\leftarrow ADD(x,y,z) \text{ and } M^+(w,z) \text{ and } M^-(w,y) \end{aligned}$$

2. Eliminate the equivalence class containing a variable x if
   - x is constant.
   - The QDE includes an explicit derivative constraint $x' = \frac{d}{dt}x$.

3. Variables in the remaining equivalence classes may chatter. Only one variable in each equivalence class needs a HOD constraint.

The ability of this algorithm to identify exactly the chattering variables is limited by the ability of an algebraic manipulator to recognize expressions that imply monotonic function constraints. If some complex expression implying $equiv(x,y)$ goes unrecognized, then the algorithm might determine that x does not chatter, but leave y unnecessarily on the list of potentially chattering variables. Even in such a case, the derivation and application of an unnecessary HOD constraint for y has a negligable effect on the performance of the algorithm.

It is also possible for the QSIM model-builder to assert explicitly which variables require higher-order derivative constraints. Statistics on the branching behavior of a simulation tree can be automatically collected to guide these assertions.

2.4 Applying the Higher-Order Derivative Constraint

The QSIM qualitative simulation algorithm operates by proposing all possible qualitative state transitions, then filtering out those that are inconsistent with available information.

Definition 3 *A filter on a set of candidates is* conservative *if it only filters out candidates that are provably inconsistent.*

As long as each filter is conservative, the algorithm preserves the guarantee that all real behaviors are predicted [Kuipers, 1986]. The higher-order derivative constraint is applied within this framework to filter out certain sequences of qualitative states. As we shall see (section 4), the HOD constraint may fail to be conservative in the presence of M^+ or M^- constraints.

Figure 3 shows which sequences of states are consistent, and which can be filtered out, given an unambiguously determined sign for $var''(t)$ or $var'''(t)$. There are two times at which the HOD constraint can be applied: when the critical point is being generated (the pre-filter), and when its successors are being generated (the post-filter).

The behavior in figure 2a, in which $var(t)$ becomes constant over an interval, is filtered out by the analytic-function constraint (Appendix A): if $var(t)$ is constant over any interval, it must be constant everywhere.

Proposition 1 *If $v(t)$ is a non-constant analytic function in the neighborhood of $t = t_i$, where $v'(t_i) = 0$, then Figure 3 shows which sequences of qualitative directions of change are consistent (or inconsistent) with knowledge of the signs of $v''(t_i)$ and $v'''(t_i)$.*

Proof: Since we are assuming that a variable $v(t)$ is analytic around a critical point t_i, in the neighborhood of t_i, the qualitative properties of v are determined by the first non-zero terms of the Taylor series:

$$v(t) \approx v(t_i) + v'(t_i)(t - t_i) + \frac{v''(t_i)}{2}(t - t_i)^2 + \frac{v'''(t_i)}{3!}(t - t_i)^3.$$

At a critical point, $v'(t_i) = 0$, if $v''(t_i) \neq 0$,

$$v(t) \approx v(t_i) + \frac{v''(t_i)}{2}(t - t_i)^2.$$

In this case, the qualitative behavior of $v(t)$ is that of t^2; that is, $dec \rightarrow std \rightarrow inc$ or $inc \rightarrow std \rightarrow dec$.

Where $v''(t_i) = v'(t_i) = 0$, but $v'''(t_i) \neq 0$,

$$v(t) \approx v(t_i) + \frac{v'''(t_i)}{3!}(t - t_i)^3,$$

so the qualitative behavior of $v(t)$ is that of t^3; that is, $inc \to std \to inc$ or $dec \to std \to dec$. ∎

This leaves us with the problem of determining the sign of $v''(t_i)$, and perhaps $v'''(t_i)$, from information in the QDE and the qualitative behavior up to t_i. If v is a variable appearing in an *ordinary* differential equation, its higher-order derivatives can be derived by repeatedly differentiating the original equation [de Kleer and Bobrow, 1984]. However, when dealing with incomplete knowledge, as represented by a *qualitative* differential equation, the problem becomes more difficult. Because of the assumptions required to derive higher-order derivatives in the presence of incompletely known monotonic function constraints (discussed in more detail in section 4), our implementation of the HOD constraint is restricted to second- and third-order derivatives.

2.5 Deriving an Expression for $sd2(var, t)$

As we have seen, chattering arises because the qualitative representation explicitly describes the magnitude of $var(t)$ and the sign of its first derivative, $qdir(var, t)$, but not the signs of its higher derivatives. However, the QDE provides a set of algebraic and differential constraints that can be used to solve for $sd2(var, t)$ in terms of values which *are* explicitly represented.

An explicit expression for $sd2(x, t_i)$ valid for t_i such that $qdir(x, t_i) = 0$ is found using a limited algebraic manipulator that searches a space of expressions generated by equivalence-preserving transformation rules. The following list illustrates the essential rules; the complete set is given in Appendix B.

$$\begin{aligned}
M^+(x, y) &\rightarrow [sd2(x, t_i) = sd2(y, t_i)] \\
x = y + z &\rightarrow [sd2(x, t_i) = sd2(y, t_i) + sd2(z, t_i)] \\
constant(x) &\rightarrow [sd2(x, t_i) = 0] \\
y = \frac{d}{dt}x &\rightarrow [sd2(x, t_i) = qdir(y, t_i)] \\
chattering_variable(x) &\rightarrow [qdir(x, t_i) = 0]
\end{aligned}$$

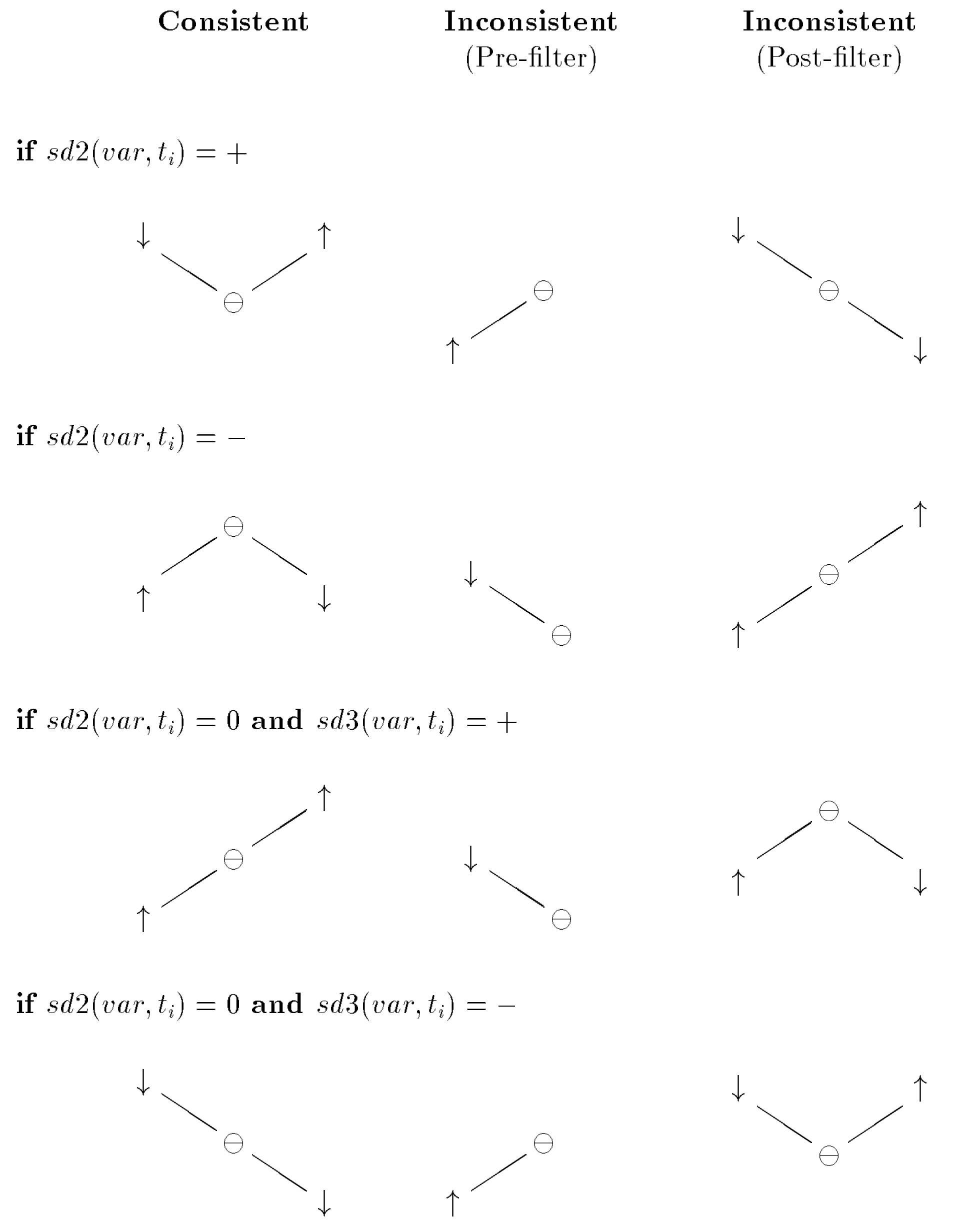


Figure 3: Consistent and Inconsistent sequences of qualitative values. In the neighborhood of a time-point t_i such that $v'(t_i) = 0$, knowledge of the signs of $v''(t_i)$ and $v'''(t_i)$ can be used to determine sequences of qualitative states that are inconsistent, and can therefore be filtered out.

Proposition 2 *Suppose a chattering variable z has a critical point at t_i, i.e. $sd1(z, t_i) = 0$. If we assume that, for any variables x and y,*

$$M^+(x, y) \to [sd2(x, t_i) = sd2(y, t_i)],$$

then each transformation in Appendix B is validity-preserving when applied at t_i.

Using this Proposition, we derive an expression for $sd2(z, t_i)$ by searching a space of expressions produced by sequences of transformations from Appendix B (along with validity-preserving algebraic simplification rules). The goal of the search is an expression that can be evaluated using the QSIM description of $State(t_i)$; i.e. no $sd2$ terms. Even then, of course, evaluation of the curvature expression may be ambiguous.

The "sign-equality" assumption embedded in this proposition is critical to higher-order derivative reasoning in the face of unknown monotonic function constraints. In section 4, we will examine this assumption in more detail, showing how to prove it is valid, and the circumstances under which it is violated.

Proof of the Proposition: The transformations involving monotonic function constraints are a restatement of the sign-equality assumption. The rule that, where z is the chattering variable, $qdir(z) \to 0$, is valid because the rules are only applied at a critical point of the chattering variable.

The addition transformation requires a bit of reflection. In the algebra of signs, the transformation $[A + B] \to [A] + [B]$ preserves validity, but may yield a weaker description when $[A] = -[B]$, because $[A + B]$ has some definite sign, while $[A] + [B] = [?]$. This allows us to conclude:

$$sd2(x, t) = [x''(t)] = [y''(t) + z''(t)] \to [y''(t)] + [z''(t)] = sd2(y, t) + sd2(z, t).$$

The remainder of the transformation rules are straight-forward consequences of the addition transformation, the identity $[A \cdot B] = [A] \cdot [B]$, and the rules for differentiation. ∎

A simple algebraic manipulator based on the rules presented in this section and in Appendix B has been adequate for the examples presented in this paper, and many others. However, manual derivation of curvature constraint expressions can apply substitutions and other algebraic methods that are beyond the power of this simple program (cf. [Dalle Molle, 1989b]). The QSIM program allows the user to assert curvature constraint expressions explicitly. The algebraic manipulator is independent of the rest of the qualitative simulator, so derivation of curvature constraint expressions could be made more powerful by incorporating a more powerful algebra package such as Macsyma [Macsyma, 1988], Mathematica [Wolfram, 1988], or MINIMA [Williams, 1988] (which is built on top of Macsyma).

2.6 Determining the Value of $sd3(var, t)$

Consideration of the second derivative allows many mechanisms, such as the two-tank cascade, to be simulated that would otherwise have been intractable. However, there are also situations where $sd2(var, t) = 0$, so the third- or higher-order derivative is necessary to apply the HOD constraint. Important examples of this are the cascaded systems of three or more tanks, for which spurious behaviors are generated when only the second-order derivative is considered, but which yield unique predictions when third-order derivatives are taken into account.

While it would be possible to construct a table of transformations for $sd3(var, t)$ analogous to the one for $sd2(var, t)$, this table would be quite complex, and turns out to be unnecessary. We may exploit the fact that $sd3(var, t_i)$ is only needed when $sd2(var, t_i) = sd1(var, t_i) = 0$. $sd3(var, t)$ can be evaluated as the derivative of the expression derived and stored for $sd2(var, t)$:

$$sd3(var, t) = \frac{d}{dt} sd2(var, t)$$

Inspection of the algebraic transformations in Appendix B reveals that the expressions that can be derived for $sd2(var, t)$ have a very restricted form. This allows us to evaluate the derivative of the expression stored for $sd2(x, t)$ using the following transformations:

$$\begin{aligned}
\frac{d}{dt}\langle number \rangle &= 0 \\
\frac{d}{dt} x &= qdir(x) \\
\frac{d}{dt}(x + y) &= \frac{dx}{dt} + \frac{dy}{dt} \\
\frac{d}{dt}(x - y) &= \frac{dx}{dt} - \frac{dy}{dt} \\
\frac{d}{dt}(x * y) &= y\frac{dx}{dt} + x\frac{dy}{dt}
\end{aligned}$$

$$\frac{d}{dt}(x/y) = \frac{1}{y}\frac{dx}{dt} - \frac{x}{y^2}\frac{dy}{dt}$$
$$\frac{d}{dt}sd1(x) = sd2(x)$$

There are two ways to evaluate terms of the form $sd2(x)$ resulting from these transformations. If the derivative x' of such a variable x is explicitly represented in the QDE, then $sd2(x,t) = sd1(x,t)$. Otherwise, if the curvature expression $sd2(x,t)$ was previously asserted or derived for x, it can simply be retrieved.

The rationale for the last rule is somewhat subtle, since the expression stored for $sd2(x,t_i)$ is based on the assumption that $sd1(x,t_i) = 0$. When evaluating $sd3(y,t_i)$, where x and y are different variables, we may assume that $sd2(y,t_i) = sd1(y,t_i) = 0$, but it is not clear that we may safely assume that $sd1(x,t_i) = 0$.

- Suppose we are attempting to evaluate
$$sd3(y,t_i) = \frac{d}{dt}sd2(y,t_i)$$
and we encounter the term $sd1(x,t)$ in the expression stored for $sd2(y)$, where x and y stand for different variables in the QDE.

- We only evaluate $sd3(y,t_i)$ when $sd2(y,t_i) = 0$. This will let us draw conclusions about the signs of terms embedded in the expression for $sd2(y,t_i)$.

- Since curvature expressions are evaluated using an algebra of signs, if the expression $A + B$ evaluates to 0, it must be because $A = 0$ and $B = 0$. (In the algebra of signs it is consistent to have $A+B = 0$ when $A = +$ and $B = -$, but in that case the evaluation of $A + B$ would have been indeterminate, not zero.)

- The same rationale applies to $A - B$ and A/B, since inspection of the rules in Appendix B reveals that a $sd1(x,t)$ term can only appear in the numerator of a quotient, and not at all in an exponential A^n. (Exponentials only arise in the quotient rules.)

- If a product $A * sd1(x,t_i)$ evaluates to zero, either $A = 0$ or $sd1(x,t_i) = 0$. Note that $sd1(x,t_i) = qdir(x,t_i)$, so its value is explicitly available in the QSIM representation of $State(t_i)$. The product rule gives us
$$\frac{d}{dt}(A * sd1(x,t_i)) = sd1(x,t_i) * \frac{d}{dt}A + A * \frac{d}{dt}sd1(x,t_i).$$

- If $sd1(x, t_i) = 0$, this is the assumption under which $sd2(x, t_i)$ was derived, so the rule $\frac{d}{dt} sd1(x, t_i) = sd2(x, t_i)$ is legitimate.
- If $sd1(x, t_i) \neq 0$ then $A = 0$, so the value (and validity) of the $\frac{d}{dt} sd1(x, t_i)$ term resulting from the product rule is irrelevant. The same reasoning applies to a product $A * B$, where $sd1(x, t_i)$ is embedded within A or B.

We summarize this discussion as the following Proposition.

Proposition 3 *Under the assumption that $M^+(x, y)$ implies that $sd2(x, t_i) = sd2(y, t_i)$ and $sd3(x, t_i) = sd3(y, t_i)$, the transformations applied in evaluating $sd3(x, t_i)$ are all validity-preserving.*

Thus, the above set of rules will give us a legitimate value for $sd3(x, t_i)$, modulo the sign-equality assumption, to be discussed in section 4.

3 Examples: Two- and Three-Tank Cascades

3.1 Higher-Order Derivative Constraints in the Two-Tank Cascade

The system of two cascaded tanks (Figure 4) is one of the simplest to exhibit chatter.

$$\begin{aligned} A' &= in - f(A) \\ B' &= f(A) - g(B) \\ & \quad f, g \in M^+ \end{aligned}$$

Figure 1 shows one behavior of this system simulated without the HOD constraint.

3.1.1 Identifying Chattering Variables

The variables in the QDE for the two-tank cascade form equivalence classes as shown. If any variable in an equivalence class has an explicit derivative in the QDE, none of the variables in the class exhibit chatter.

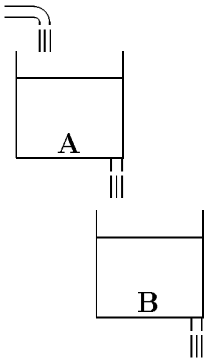
$\{in\}$	no chatter, because in is constant.
$\{A, f(A), A'\}$	no chatter, because $A' = dA/dt$.
$\{B, g(B)\}$	no chatter, because $B' = dB/dt$.
$\{B'\}$	chatters

Therefore, the variable B' (named `netflowB` in the QSIM code) chatters, so we need to apply the HOD constraint (Figure 1).

3.1.2 Deriving the Curvature Constraint

The derivation of the curvature constraint is the following. Recall that we only apply the value of $sd2(B')$ when $qdir(B') = 0$.

$$\begin{aligned} sd2(B') &= sd2(f(A)) - sd2(g(B)) \\ &= sd2(A) - sd2(B) \\ &= qdir(A') - qdir(B') \\ &= qdir(A') \end{aligned}$$



$$
\begin{aligned}
A' &= in - f(A) \\
B' &= f(A) - g(B) \\
& \quad f, g \in M^+
\end{aligned}
$$

```
(define-QDE Two-Tank-Cascade
  (quantity-spaces
    (inflowa        (0 inf))                    ; in
    (amounta        (0 inf))                    ; A
    (outflowa       (0 inf))                    ; f(A)
    (netflowa   (minf 0 inf))                   ; A'
    (amountb        (0 inf))                    ; B
    (outflowb       (0 inf))                    ; g(B)
    (netflowb   (minf 0 inf)))                  ; B'
  (constraints
    ((M+ amounta outflowa)          (0 0) (inf inf))
    ((add outflowa netflowa inflowa))
    ((d/dt amounta netflowa))
    ((M+ amountb outflowb)          (0 0) (inf inf))
    ((add outflowb netflowb outflowa))
    ((d/dt amountb netflowb))
    ((constant inflowa)))
```

Figure 4: The two-tank cascade and its QDE model in algebraic and QSIM forms.

In terms of the QSIM variables,

$$qdir(netflowB, t_i) = 0 \rightarrow [sd2(netflowB, t_i) = qdir(netflowA, t_i)].$$

.

3.1.3 Applying the Curvature Constraint

Consider the behavior of $netflowB$ illustrated in Figure 1, and consider the critical points at t_1 and t_2, where $qdir(netflowB, t) = 0$.

- At t_1, we know that $sd2(netflowB, t_1) = qdir(netflowA, t_1) = -$, so the concave-down behavior at $netflowB(t_1)$ is acceptable (Figure 3).
- At t_2, we know that $sd2(netflowB, t_2) = qdir(netflowA, t_2) = -$, but the predicted behavior of $netflowB(t_2)$ is concave-up, so this behavior is inconsistent (Figure 3, pre-filter).

With the curvature constraint, instead of the intractable branching of Figure 1, QSIM predicts the two-tank cascade to have a unique qualitative behavior (Figure 5).

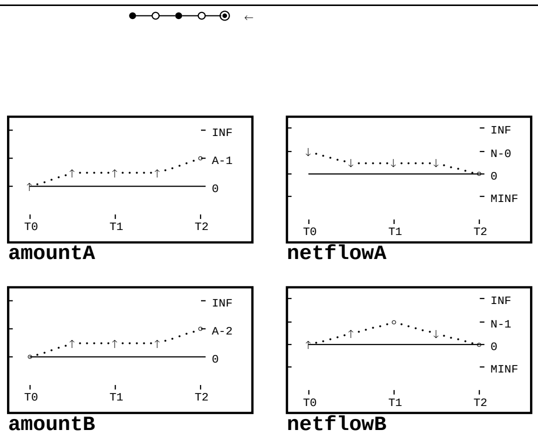


Figure 5: Unique qualitative behavior predicted for the two-tank cascade with $sd2$ constraint.
Only a single behavior out of the intractably branching tree in figure 1 satisfies the $sd2$ constraint

$$qdir(netflowB, t_i) = 0 \rightarrow [sd2(netflowB, t_i) = qdir(netflowA, t_i)].$$

3.2 Third-Order Derivatives: the Three-Tank Cascade

The three-tank cascade is structurally similar to the two-tank cascade, but it is no longer possible to eliminate all spurious behaviors with the second-order derivative alone. We will require a third-order derivative. Fortunately, second- and third-order derivatives are adequate for cascades of any length.

In algebraic form, the QDE for the three-tank cascade is:

$$\begin{aligned} A' &= in - f(A) \\ B' &= f(A) - g(B) \\ C' &= g(B) - h(C) \\ & \quad f, g, h \in M^+ \end{aligned}$$

3.2.1 Identifying Chattering Variables

The equivalence classes for the variables in the three-tank cascade are the following.

$\{in\}$	no chatter because in is constant.
$\{A, f(A), A'\}$	no chatter, because $A' = dA/dt$.
$\{B, g(B)\}$	no chatter, because $B' = dB/dt$.
$\{B'\}$	chatters
$\{C, h(C)\}$	no chatter, because $C' = dC/dt$.
$\{C''\}$	chatters

Thus, we will need expressions for higher-order derivatives of B' and C''.

3.2.2 Deriving and Applying Curvature Constraints

Using the same method as for the two-tank cascade, we derive expressions for $sd2(B')$ and $sd2(C'')$:

$$\begin{aligned} sd2(netflowB) &= qdir(netflowA) \\ sd2(netflowC) &= qdir(netflowB) \end{aligned}$$

Application of these constraints eliminates many branches, but still leaves two spurious behaviors. For example, in the two behaviors shown in figure 6, the critical points at $netflowC(t_1)$ are not possible in actual behaviors, but could not be eliminated by $sd2$ alone, because

$$sd2(netflowC, t_1) = qdir(netflowB, t_1) = 0.$$

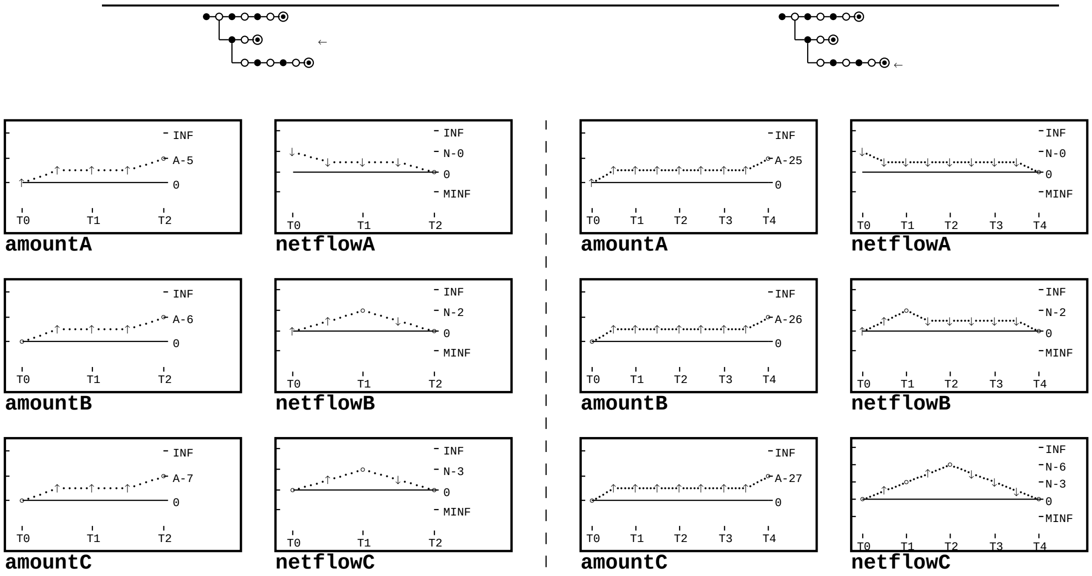


Figure 6: The three-tank cascade with $sd2$ constraints.

After deriving and applying the $sd2$ constraints

$$qdir(netflowB, t_i) = 0 \quad \rightarrow \quad [sd2(netflowB, t_i) = qdir(netflowA, t_i)]$$
$$qdir(netflowC, t_i) = 0 \quad \rightarrow \quad [sd2(netflowC, t_i) = qdir(netflowB, t_i)]$$

intractable branching has been prevented, but two of the three predicted behaviors are still spurious. They cannot be filtered because $sd2(netflowC, t_1) = 0$.

3.2.3 Evaluating the $sd3$ Constraint

We determine $sd3(netflowC, t)$ by differentiating the expression stored for $sd2(netflowC, t)$.

$$\begin{aligned} sd3(netflowC) &= \frac{d}{dt} sd2(netflowC) \\ &= \frac{d}{dt} qdir(netflowB) \\ &= sd2(netflowB) \\ &= qdir(netflowA) \end{aligned}$$

Thus, in the two spurious behaviors shown in figure 6, $sd3(netflowC, t_1) = qdir(netflowA, t_1) = -$. Consulting the table of acceptable qualitative transitions in figure 3 demonstrates that both behaviors in figure 6 will be filtered out by the pre-filter. Figure 7 then shows the single behavior resulting from simulation using both $sd2$ and $sd3$ in the HOD constraint.

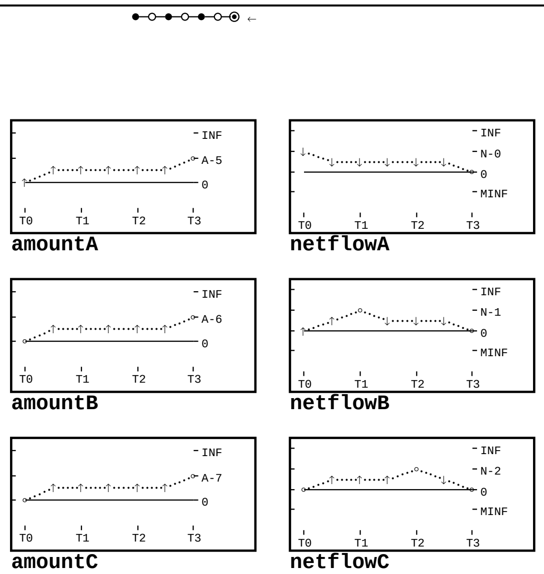


Figure 7: Unique qualitative behavior predicted for the three-tank cascade with $sd2$ and $sd3$ constraints.

By deriving and applying the constraint for $sd3$,

$$sd2(netflowC, t_i) = qdir(netflowC, t_i) = 0 \rightarrow [sd3(netflowC, t_i) = qdir(netflowA, t_i]$$

the two spurious behaviors in figure 6 are filtered out, and this unique prediction remains.

4 Monotonic Function Constraints

A major strength of qualitative reasoning is the ability to obtain useful predictions in the face of incomplete knowledge of the structure of a mechanism. A key method for expressing this knowledge in QSIM is the monotonic function constraint, allowing one to assert that two variables are related by some function which is only known to be monotonically increasing or decreasing. For example, the following constraints could appear in QDE models of a liquid-tank or a spring, respectively.

$$M^+(\textit{liquid-level}, \textit{outflow-rate})$$

$$M^-(\textit{spring-displacement}, \textit{restoring-force})$$

When it is desirable to provide a name for a monotonic function, we may use an alternate notation:

$$\textit{outflow-rate} = f(\textit{liquid-level}), \ f \in M^+$$

$$\textit{restoring-force} = -g(\textit{spring-displacement}), \ g \in M^+$$

The function $f \in M^+$ is known to satisfy $f' > 0$ everywhere on the interior of its domain, so it is strictly monotonically increasing [Kuipers, 1986].[2] However, f'' is unspecified. Monotonic function constraints are useful for expressing incomplete knowledge, but they raise important problems when reasoning about higher-order derivatives [Crawford, Farquhar & Kuipers, 1990]..

4.1 The Sign-Equality Assumption

The constraint $M^+(x, y)$ means that there is some $f \in M^+$ such that for all t, $y(t) = f(x(t))$. (M^- constraints are handled similarly, with certain terms negated.) We can take the first derivative of this expression, to get

$$y'(t) = f'(x(t)) * x'(t).$$

[2]The QSIM constraint $M^+(x, y)$ is closely related to, but not identical to, the confluence $\partial x = \partial y$ [de Kleer and Brown, 1984] or the qualitative proportionality $y \,\alpha_{Q+}\, x$ [Forbus, 1984]. The confluence is weaker than the M^+ constraint in that it does not imply that there is a function underlying the relationship. Qualitative proportionality is an "open-world" assertion that must be combined with all other influences on the same variables to produce a constraint [Crawford, Farquhar & Kuipers, 1990].

Since $f' > 0$, this tells us that $[y'(t)] = [x'(t)]$, or $qdir(y,t) = qdir(x,t)$. However, f'' is unspecified, so the second-derivative relationship is weaker:

$$y''(t) = f'(x(t)) * x''(t) + f''(x(t)) * (x'(t))^2.$$

The rule for solving for $sd2(var, t)$ in the presence of monotonic function constraints relies on the *sign-equality assumption*, that:

$$[y''(t)] = [x''(t)].$$

The sign-equality assumption is correct whenever $x''(t)$ and f'' have the same sign, or when $f \in M^+$ is linear, so $f'' = 0$. Because of the role of the sign-equality assumption, the higher-order derivative constraint is potentially not a conservative filter, when the QDE includes monotonic function constraints.

Proposition 4 *If every monotonic function constraint $M^+(x,y)$ in a QDE satisfies $sd2(x,t_i) = sd2(y,t_i)$ at a qualitative time-point t_i, then filtering according to the $sd2$ constraint is conservative at t_i.*

Proposition 5 *Suppose that a QDE contains a monotonic function constraint $M^+(x,y)$ representing an unknown function $f \in M^+$ such that $y = f(x)$, and suppose the $sd2$ constraint is being applied at qualitative time-point t_i. Then $sd2(x,t_i) = sd2(y,t_i)$ in case* any *of the following conditions hold:*

1. *The function $f \in M^+$ is linear;*
2. $x'(t_i) = 0$;
3. $[x''(t_i)] = [f''(x(t_i))]$;
4. $[y''(t_i)] = -[f''(x(t_i))]$;
5. $[x''(t_i)] = -[f''(x(t_i))]$ *and* $|f''(x(t_i))(x'(t_i))^2| < |f'(x(t_i))x''(t_i)|$.

Proof: $y''(t_i) = f'(x(t_i)) * x''(t_i) + f''(x(t_i)) * (x'(t_i))^2$, and $f' > 0$, and $(x'(t))^2 \geq 0$. ∎.

The first four conditions in this Proposition rely on the availability of additional qualitative knowledge, such as the sign of f''. The fifth condition is a quantitative criterion, and cannot be established using a purely qualitative description of a system. Kuipers and Berleant (1988) present a method for reasoning with incomplete quantitative information in a qualitative framework. Their method can be extended to evaluate such a condition, where quantitative bounds on f', f'', x', and x'' can be obtained.

One can, however, construct examples where the $f''(x(t)) * (x'(t))^2$ term makes a significant contribution to the sign relationship, so the sign-equality assumption is violated.

4.2 Example: Violating The Sign-Equality Assumption

We have already seen, in figure 5, the predicted qualitative behavior of a two-tank cascade. Notice that $netflowB(t)$ rises monotonically from zero to its maximum value, then falls monotonically back to zero. Suppose we consider an actual pair of tanks such that the upper tank has a stack (fig. 8a), so that the monotonic relationship

$$outflowA = f(amountA)$$

has a sharp bend (fig. 8b). In this case, $amountA(t)$ is concave down, but f'' is large and positive, with the net result that $outflowA(t)$ is actually concave up at this point.

Numerical simulation of a model of this situation gives the behavior shown in figure 9. All variables are consistent with the qualitative prediction except for $netflowB$, which includes a significant dip.

We can perform a numerical sensitivity analysis on the curvature of $M^+(amountA, outflowA)$. As the curvature in $M^+(amountA, outflowA)$ becomes smoother, the unpredicted dip in $netflowB(t)$ becomes smaller, and the actual behavior converges to the qualitative prediction (Figure 10). Traditional Taylor series methods make it possible to estimate the magnitude of the error as a function of the magnitudes of the derivatives of the monotonic function, but they are outside the scope of this paper.

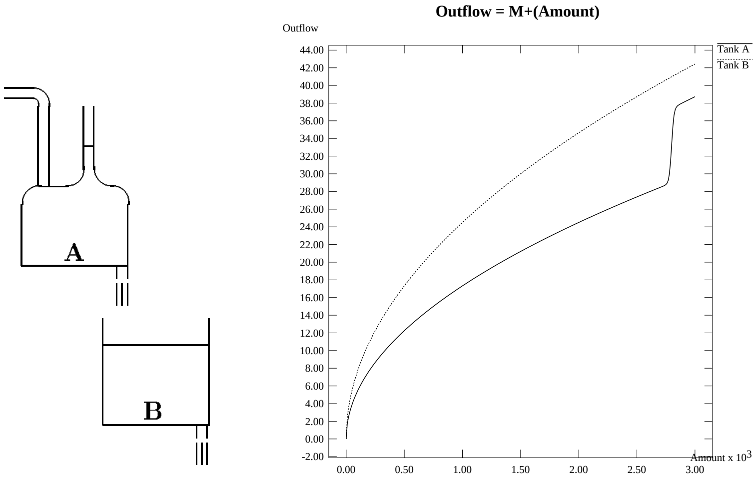


Figure 8: Two cascaded tanks where $outflowA = M^+(amountA)$ has a sharp bend.

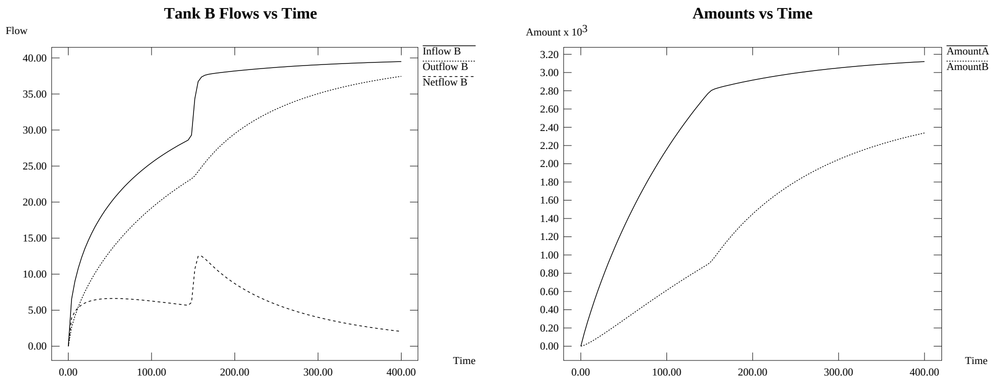
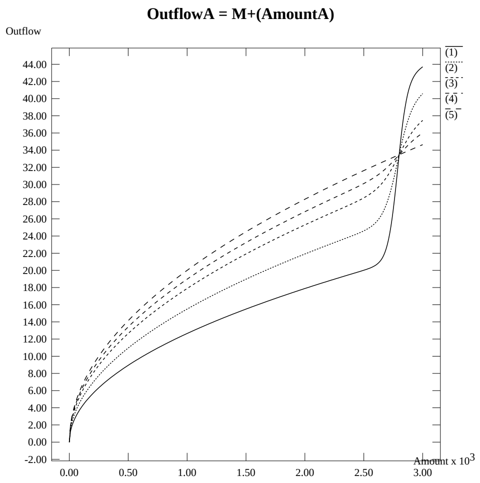


Figure 9: Numerical simulation of the two-tank cascade.

Contrary to the qualitative prediction in figure 5, $netflowB(t)$ includes a pronounced dip and rise. If the assumption of a single maximum for $netflowB(t)$ were used at $t = 150$ to predict the minimum time until $amountB = 2000$, a significant error would result.



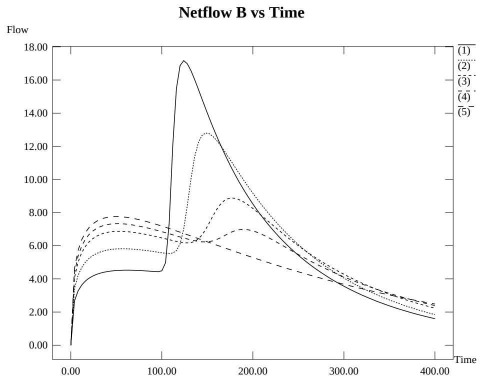


Figure 10: Sensitivity to curvature of $outflowA = M^{+}(amountA)$.

As the curvature in $M^{+}(amountA, outflowA)$ becomes smoother, the unpredicted dip and rise in $netflowB(t)$ becomes smaller, and the actual behavior converges to the qualitative prediction in figure 5.

4.3 Avoiding Prediction Failure

One of the attractive features of qualitative simulation is the ability to predict all possible behaviors consistent with incomplete knowledge. Thus, the prospect of failing to predict actual behaviors, due to the use of a non-conservative filter, is quite troubling. However, a deeper analysis of these prediction failures demonstrates that, while the phenomenon is real, there are a number of effective strategies for avoiding or minimizing problems due to it.

- These prediction failures only occur in the presence of monotonic function constraints. Although avoiding monotonic function constraints sacrifices an important part of the expressive power for incomplete knowledge, qualitative simulation of *ordinary* differential equations can still provide valuable insight into the set of all possible behaviors of a system.

- Prediction failures arise because the derived higher-order derivative constraint eliminates a genuine behavior of a chattering variable. The qualitative predictions about non-chattering variables are completely reliable. If a variable $V(t)$ is predictively important, one may include an explicit variable for its derivative $V'(t)$, along with derived constraints on that variable. The problem of chatter will still need to be solved at the level of $V'(t)$, and may be more difficult to solve, but predictions about the behavior of $V(t)$ will be reliable.

- It may be possible to extend the representation for higher-order derivative expressions to record their dependency on monotonic functions. If qualitative or quantitative information is available about the slopes and curvatures of monotonic functions, the possibility and magnitude of violations of the sign-equality assumption can be determined [Kuipers and Berleant, 1988], to more fully exploit the conditions in Proposition 5.

An alternate method of eliminating the phenomenon of chatter is to change the level of qualitative description, accepting a weaker description of the predicted behaviors, and a lesser degree of filtering of spurious predictions, in return for the guarantee that all real behaviors are predicted. This is the method we turn to in the next section.

5 Changing Level of Description

In this section, we develop an alternate solution to the problem of unconstrained, chattering variables. The two solutions each have their own strengths and weaknesses, and each technique suggests a direction for further developments in qualitative reasoning methods.

Consider the two cascaded tanks (figure 4). As we have seen, the chattering variable

$$netflowB(t) = outflowA(t) - outflowB(t)$$

is the difference between two other variables, both of which are increasing monotonically with time in this situation (figure 1b). Thus, the direction of change, $qdir(netflowB, t)$, is constrained only by continuity. In a particular instance of this model, the details of how $netflowB(t)$ behaves are determined by the detailed behavior of $outflowA(t)$ and $outflowB(t)$. These, in turn, are determined by the particular monotonic functions described by the constraints,

$$\begin{aligned} outflowA(t) &= M^{+}(amountA(t)) \\ outflowB(t) &= M^{+}(amountB(t)). \end{aligned}$$

Depending on how the two monotonic functions interact, the actual behavior of $netflowB(t)$ may rise and fall any number of times. I.e., the "spurious" prediction in figure 1b accurately describes the behavior in figure 9 of a real system. Therefore, we must accept the conclusion that the intractably branching tree of predicted behaviors represents an infinite collection of *real* possibilities: the set of all possible behaviors violating the sign-equality assumption.

5.1 Collapsing Descriptions

However, even though the behaviors are genuine, and qualitatively distinct, the distinctions between them may be uninteresting to a problem-solver. An effective approach in this situation is to adopt an alternate level of description that collapses an infinite set of possible behaviors into a single description, while preserving validity.

In the case of the two-tank cascade, $netflowB(t)$ is the chattering variable, and the distinctions among behaviors can be attributed to changes in $qdir(netflowB, t)$. If we replace the distinctions between *inc*, *std*, and *dec*

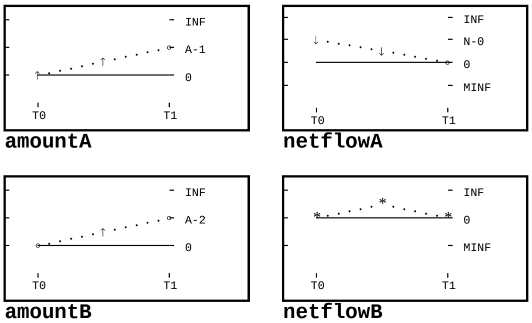


Figure 11: Ignoring distinctions in $qdir(netflowB, t)$ gives a unique behavior for the two-tank cascade.
The symbol * represents the *ign* qdir.

when describing the direction of change of $netflowB$, with a single value ign (for "ignore"), then the infinite, intractably branching tree of behaviors collapses into a single finite behavior (fig. 11).

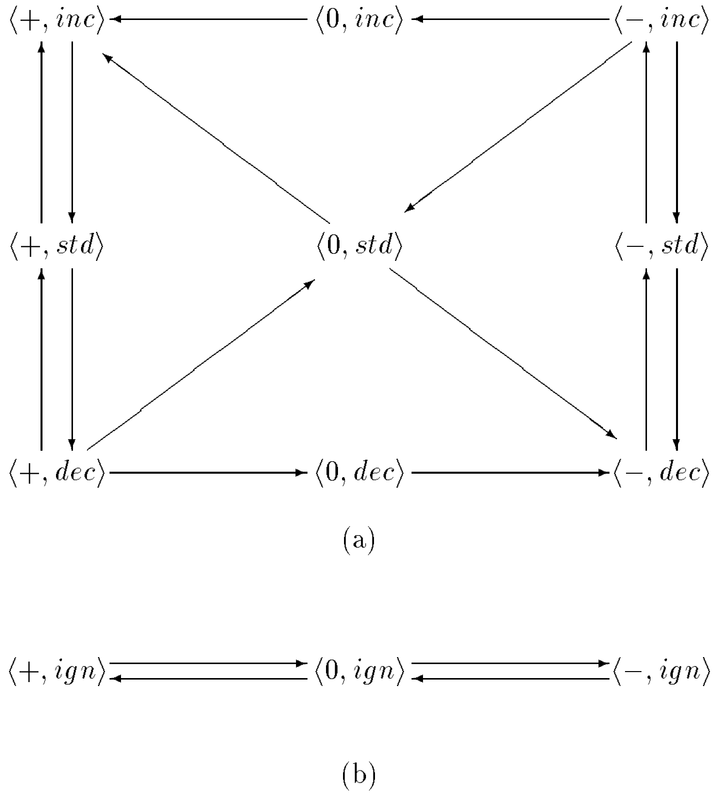
In order to eliminate chattering, this "ignore-qdir" description must be applied to *every* variable in the equivalence classes defined in section 2.1.

Comparing figure 11 with figure 5, we see that the description captures many of the same qualitative features. However figure 11 represents a weaker description of the behavior of $netflowB(t)$ than figure 5 does. The qualitative description that $netflowB(t) = \langle (0, \infty), ign \rangle$ for $t \in (t_0, t_1)$ is consistent with any number of dips and oscillations, as long as they don't reach the endpoints of the interval. The prediction in figure 5 is significantly stronger.

Figure 12 uses a simplified $\{+, 0, -\}$ quantity space with a single landmark at 0 to illustrate the qualitative transitions possible during chatter. Changing the level of description collapses an infinite family of behaviors wandering among the states $\langle +, inc \rangle$, $\langle +, std \rangle$, and $\langle +, dec \rangle$, into the single qualitative state description $\langle +, ign \rangle$.

5.2 Verifying Viability

Unfortunately, if we simply collapse the transitions in Fig. 12a to the simpler set in Fig. 12b, we lose an important source of constraint: the derivatives of



- (a) The full qualitative transition graph is adequate to capture continuity constraints, but permits "chattering" behaviors.
- (b) The collapsed transition graph, ignoring direction of change, eliminates chatter, but fails to detect discontinuous change.

Figure 12: Transition graphs for a single unconstrained qualitative variable around the landmark 0.

variables must change continuously. For example, although the transition

$$\langle +, ign\rangle \longrightarrow \langle 0, ign\rangle$$

is apparently consistent (Fig. 12b), the more specific transition

$$\langle +, ign\rangle \longrightarrow \langle 0, inc\rangle$$

is inconsistent with the requirement that variables in QDEs be continuously differentiable.

To recapture the constraint that the derivative of a variable must change continuously, we apply a global *satisfiability filter* to each state where the ign direction of change was used. The satisfiability filter determines whether there is a complete, consistent state in which each occurrence of ign is replaced by one of $\{inc, std, dec\}$, and which is a consistent successor of the previous state. The satisfiability filter is clearly conservative.

Proposition 6 *The set of behaviors predicted by QSIM, applying the "ignore-qdir" description to any subset of variables in the QDE, includes every consistent behavior predicted by QSIM using the standard qualitative description.*

Proof sketch: We know that the set of qualitative value transitions provided in [Kuipers, 1986] includes every possible transition. As illustrated by figure 12, the set of possible transitions under the ignore-qdir description encompasses each of those transitions, so all possible qualitative state changes are proposed. Since the satisfiability filter eliminates only inconsistent states, every actual behavior must remain. ∎

The satisfiability filter is a weaker constraint than simulation with the larger set of distinctions, $\{inc, std, dec\}$. For example, it may be possible for a sequence of qualitative states

$$S1 \longrightarrow S2 \longrightarrow S3$$

to survive the satisfiability filter because one set of substitutions is consistent with $S1 \rightarrow S2$, while another is consistent with $S2 \rightarrow S3$, although no one set of substitutions is consistent with both transitions.

5.3 Strengths and Weaknesses

Changing level of description has two advantages over the explicit higher-order derivative constraint:

- It makes no assumptions about the M^+/M^- functions, and thus preserves the desirable property that all real behaviors are predicted.
- It can be implemented within the constraint-filtering computational framework of existing qualitative simulation algorithms, rather than requiring a possibly elaborate algebraic manipulation package (see Appendix B).

However, there are two significant disadvantages as well.

- The coarser level of description makes it impossible to derive information about higher-order derivatives that could be used to filter out genuinely spurious behaviors [Dalle Molle, 1989b]. Figure 13 shows such an example.
- The coarser level of description produces a weaker prediction, and hence is less useful for explaining observations or for hypothesis-testing.

Thus, the choice of method for handling chatter depends on which variables must be described to what degree of detail. In complex models, it may be appropriate to determine higher-order derivative constraints for certain variables, while ignoring qdirs on others [Dalle Molle, 1989b].

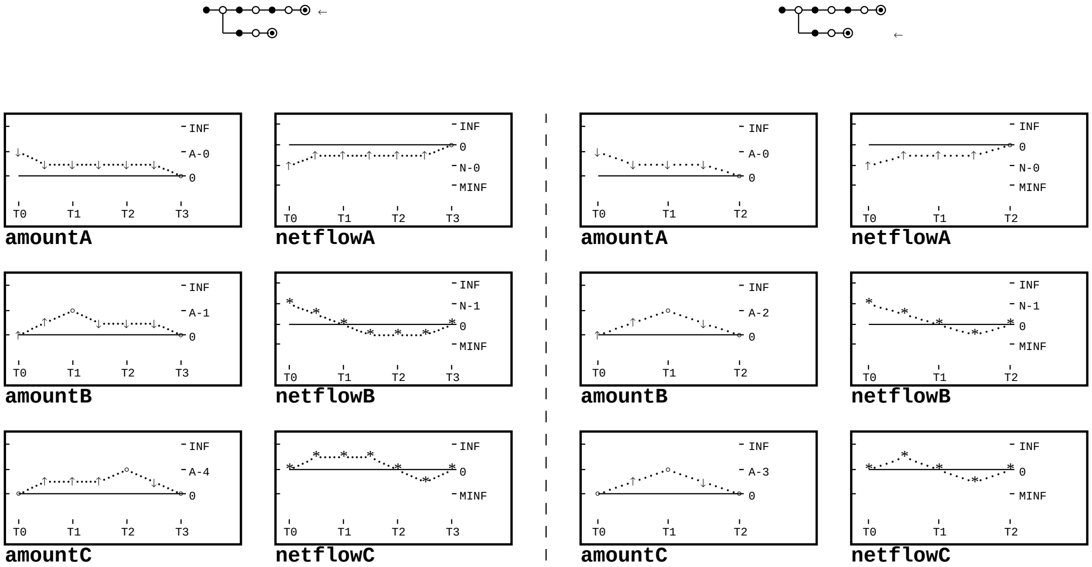


Figure 13: Our behavior is genuine; the other is not.
Consider the three-tank cascade, initialized with tank A filled, and draining through tanks B and C until all tanks are empty. It is not possible for both $B(t)$ and $C(t)$ to have critical points at the same time, $t_i$but the information required to filter out this possibility is not available when ignoring qdirs.

6 Conclusions

As we have seen, an important source of intractable branching in qualitative simulation is lack of constraint on the direction of change of certain variables, due to lack of information about the higher-order derivatives of those variables.

One method for eliminating this type of branching is to derive and apply the required information about higher-order derivatives: the *HOD constraint*. It is possible to do this while focusing attention on the higher-order derivatives only at those isolated points where branching takes place. The disadvantage of this approach is that it requires certain assumptions about the behavior of monotonic function constraints which may not, in general, be warranted. This sign-equality assumption means that certain qualitative behaviors may be filtered out, in spite of being genuine possibilities. A useful direction for future research would be the determination of when the prediction is quantitatively "close enough" to the actual behavior.

A second method for eliminating this branching is to collapse the descriptions of certain directions of change, to avoid representing unimportant distinctions. This method avoids reliance on added assumptions about monotonic function constraints. However, this conservative approach produces a slightly weaker description of the predicted behavior, and the ability to filter out spurious predictions is reduced.

Thus, we observe another instance of the classic trade-off between generality and power (or false-negative versus false-positive error rates). Which method is most appropriate depends on the details of the pragmatic context within which the simulation is being used. For example, one must ask how much knowledge is actually available to bound the curvature of unknown monotonic functions, and how serious a deviation between prediction and observation (e.g. the "dip" in fig. 9) can be tolerated at what cost.

These higher-order derivative constraint methods have been sufficient to allow tractable predictions of the possible behaviors of open and closed two-tank systems, cascades of any number of tanks, and numerous other mechanisms drawn from chemical engineering [Dalle Molle, 1989b]. These types of multi-compartment models are generic instances of such systems as chemical reaction kinetics [Dalle Molle and Edgar, 1989a], physiological mechanisms [Jacquez, 1985], ecological systems [Puccia and Levins, 1985], etc.

There are many other important qualitative mechanism models for which the higher-order derivative constraints are a necessary, but not sufficient, source of constraint to obtain a tractable behavior. For example, a damped oscillatory system such as the PI controller requires the local constraint provided by the higher-order derivative constraint, but also requires non-local constraints such as energy and system property constraints [Lee, Chiu, and Kuipers, 1987; Fouchéand Kuipers, 1990], and the non-intersection constraint in qualitative phase space [Lee and Kuipers, 1988; Struss, 1988; Dalle Molle and Edgar, 1989b].

Thus, the higher-order derivative constraints are essential pieces in the puzzle. The overall picture, at least as far as qualitative simulation goes, is approximately the following:

- Limit analysis algorithms [de Kleer and Brown, 1984; Forbus, 1984; Kuipers, 1984, 1986] predict the local transitions from one qualitative state to its immediate successors, and can be constructed to guarantee that all possible behaviors are predicted, although it is not possible to guarantee the elimination of all spurious behaviors [Struss, 1988a; Kuipers, 1988].

- Higher-order derivative constraints can be applied to eliminate an important class of intractable branching, as described in this paper and in [de Kleer and Bobrow, 1984; Williams, 1984b; Kuipers and Chiu, 1987; Chiu, 1988].

- Quantitative information, in the form of measurements or *a priori* knowledge, can be combined with qualitative predictions to determine which qualitative behaviors are consistent with the quantitative knowledge [Forbus, 1983, 1986; Kuipers and Berleant, 1988].

- Non-local constraints obtainable from the energy and system property constraints [Lee, Chiu, and Kuipers, 1987], and the qualitative phase space can eliminate other spurious behaviors [Lee and Kuipers, 1988; Struss, 1988b]. An analysis based on the Kinetic Energy Theorem [Fouchéand Kuipers, 1990] now makes it possible to produce a tractable simulation of the non-linear monotonic damped spring and hence such industrially significant mechanisms as PI controllers.

- Hierarchical decomposition methods will be usable to decompose certain complex systems into weakly-coupled subsystems [Kuipers, 1987; Simon and Ando, 1961; Iwasaki and Bhandari, 1988], ideally to the point that the subsystems are small enough for the previous methods to be effective.

- Comparative analysis methods [Forbus, 1984; Weld, 1987; Chiu and Kuipers, 1989] make it possible to determine the effects on an individual qualitative behavior of perturbations to variables, and to determine relations among "adjacent" behaviors.

As these pieces of the puzzle are filled in, we expect that qualitative simulation will be adequate for model-based reasoning about realistically complex systems in the presence of incomplete knowledge.

7 References

1. Bird, Stewart, and Lightfoot. 1960. *Transport Phenomenon.* John Wiley and Sons.

2. W. E. Boyce and R. C. DiPrima. 1969. *Elementary Differential Equations*, second edition. New York: John Wiley.

3. C. Chiu. 1988. Higher Order Derivative Constraints and a QSIM-based Total Simulation Scheme. University of Texas Artificial Intelligence Laboratory TR-88-65, Austin, Texas.

4. C. Chiu and B. Kuipers. 1989. Comparative analysis and qualitative integral representations. *Workshop on Qualitative Reasoning*, Stanford, CA, August 1989.

5. J. Crawford, A. Farquhar and B. Kuipers. 1990. QPC: a compiler from physical models into qualitative differential equations. *Proceedings of the National Conference on Artificial Intelligence (AAAI-90).* Cambridge: AAAI Press/MIT Press.

6. D. T. Dalle Molle. 1989a. Automatic Evaluation of Third-Order Constraints in Qualitative Simulation. University of Texas Artificial Intelligence Laboratory TR-89-98, Austin, Texas.

7. D. T. Dalle Molle. 1989b. *Qualitative Simulation of Dynamic Chemical Processes.* Ph.D. dissertation, Department of Chemical Engineering, University of Texas at Austin, Austin, Texas.

8. D. T. Dalle Molle and T. F. Edgar. 1989a. Qualitative modeling of chemical reaction systems. In M Mavrovouniotis (Ed.), *Artificial Intelligence Applications in Process Engineering*, Academic Press, 1989.

9. D. T. Dalle Molle and T. F. Edgar. 1989b. Qualitative simulation for process modeling and control. In *Proceedings of the American Control Conference*, 1989.

10. D. T. Dalle Molle, B. J. Kuipers, and T. F. Edgar. 1988. Qualitative modeling and simulation of dynamic systems. *Computers and Chemical Engineering* **12**: 853-866, 1988.

11. J. de Kleer and D. G. Bobrow. 1984. Qualitative reasoning with higher-order derivatives. In *Proceedings of the National Conference on Artificial Intelligence (AAAI-84).* Los Altos, CA: Morgan-Kaufman Publishers.

12. J. de Kleer and J. S. Brown. A qualitative physics based on confluences. *Artificial Intelligence* **24**: 7 - 83, (1984).

13. K. D. Forbus. 1983. Measurement interpretation in qualitative process theory. In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence (IJCAI-83)*. Los Altos, CA: Morgan Kaufman.

14. K. D. Forbus. Qualitative process theory. *Artificial Intelligence* **24**: 85 - 168, (1984).

15. K. D. Forbus. 1986. Interpreting measurements of physical systems. In *Proceedings of the National Conference on Artificial Intelligence (AAAI-86)*. Los Altos, CA: Morgan Kaufman.

16. Y. Iwasaki and I. Bhandari. 1988. Formal basis for commonsense abstraction of dynamic systems. In *Proceedings of the National Conference on Artificial Intelligence (AAAI-88)*. Los Altos, CA: Morgan Kaufman.

17. John A. Jacquez. 1985. *Compartmental Analysis in Biology and Medicine*, second edition. Ann Arbor: University of Michigan Press.

18. B. J. Kuipers. 1984. Commonsense reasoning about causality: deriving behavior from structure. *Artificial Intelligence* **24**: 169 - 204.

19. B. J. Kuipers. 1985. The limits of qualitative simulation. In *Proceedings of the Ninth International Joint Conference on Artificial Intelligence (IJCAI-85)*. William Kaufman, Los Altos, CA.

20. B. J. Kuipers. 1986. Qualitative simulation. *Artificial Intelligence* **29**: 289 - 338.

21. B. Kuipers. 1987a. Qualitative Simulation as Causal Explanation. *IEEE Transactions on Systems, Man, and Cybernetics* **SMC-17**(3): 432-444.

22. B. Kuipers. 1987b. Abstraction by time-scale in qualitative simulation. *Proceedings of the National Conference on Artificial Intelligence (AAAI-87)*. Los Altos, CA: Morgan Kaufman.

23. B. Kuipers. 1988. The qualitative calculus is sound but incomplete: a reply to Peter Struss. *Int. J. Artificial Intelligence in Engineering* **3**(3): 170-173.

24. B. Kuipers. 1989. Qualitative reasoning: modeling and simulation with incomplete knowledge. *Automatica* **25**: 571-585.

25. B. Kuipers and D. Berleant. 1988. Using incomplete quantitative knowledge in qualitative reasoning. In *Proceedings of the National Conference on Artificial Intelligence (AAAI-88)*. Los Altos, CA: Morgan Kaufman.

26. B. Kuipers and C. Chiu. 1987. Taming intractible branching in qualitative simulation. *Proceedings of the Tenth International Joint Conference on Artificial Intelligence (IJCAI-87)*. Los Altos, CA: Morgan Kaufman.

27. W. W. Lee, C. Chiu, and B. Kuipers. 1987. Steps toward constraining qualitative simulation. University of Texas Computer Science TR-87-44.

28. W. W. Lee and B. Kuipers. 1988. Non-intersection of trajectories in qualitative phase space: a global constraint for qualitative simulation. In *Proceedings of the National Conference on Artificial Intelligence (AAAI-88).* Los Altos, CA: Morgan Kaufman.

29. Macsyma Reference Manual. 1988. Mathlab Group, Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, Massachusetts. (Commercial version available from Symbolics, Inc.)

30. Charles J. Puccia and Richard Levins. 1985. *Qualitative Modeling of Complex Systems.* Cambridge, MA: Harvard University Press.

31. P. A. Sachs, A. M. Paterson, and M. H. M Turner. 1986. Escort – an expert system for complex operations in real time. *Expert Systems* v. 3, n. 1, pp. 22-29.

32. H. A. Simon and A. Ando. 1961. Aggregation of variables in dynamic systems. *Econometrica* **29**: 111-138.

33. Peter Struss. 1988a. Mathematical aspects of qualitative reasoning *Int. J. Artificial Intelligence in Engineering* **3**(3): 156-169.

34. Peter Struss. 1988b. Global filters for qualitative behaviors. In *Proceedings of the National Conference on Artificial Intelligence (AAAI-88).* Los Altos, CA: Morgan Kaufman.

35. G. J. Sussman and R. M. Stallman. 1975. Heuristic techniques in computer-aided circuit analysis. *IEEE Trans. on Circuits and Systems*, CAS-22(11): 857-865.

36. D. R. Throop. 1989. On the applicability of the smoothness assumption to qualitative simulation. University of Texas Artificial Intelligence Laboratory TR-89-94, Austin, Texas.

37. Daniel S. Weld. 1987. Comparative Analysis. *Proceedings of the Tenth International Joint Conference on Artificial Intelligence (IJCAI-87).* Los Altos, CA: Morgan Kaufman.

38. D. S. Weld. 1988. Comparative analysis. *Artificial Intelligence* **36**: 333-374.

39. D. S. Weld. 1990. Exaggeration. *Artificial Intelligence* **43**: 311-368.

40. B. Williams. 1984. Qualitative analysis of MOS circuits. *Artificial Intelligence* **24**: 281 - 346. (a)

41. B. Williams. 1984. The use of continuity in a qualitative physics. In *Proceedings of the National Conference on Artificial Intelligence (AAAI-84)*, 350 - 354. (b)

42. Brian Williams. 1986. Doing time: putting qualitative reasoning on firmer ground. In *Proceedings of the Fifth National Conference on Artificial Intelligence (AAAI-86).* Los Altos, CA: Morgan Kaufman Publishers, pp. 105-112.

43. Brian C. Williams. 1988. MINIMA: A symbolic approach to qualitative algebraic reasoning. In *Proceedings of the Seventh National Conference on Artificial Intelligence (AAAI-88).* Los Altos, CA: Morgan Kaufman Publishers, pp. 264-269.

44. Stephen Wolfram. 1988. *Mathematica: A System for Doing Mathematics by Computer.* Addison-Wesley, Reading, Massachusetts.

A The Analytic Function Restriction

The basic limit analysis algorithms for qualitative simulation, QSIM for example, require that variables be continuously differentiable functions of time. That is, for any variable v, $v(t)$ must be continuous, and its derivative $v'(t)$ must be defined and continuous. Because of this restriction, we can depend on the qualitative description of $v(t)$ changing in orderly transitions from one state to adjacent ones (in both magnitude and direction of change).

Higher-order derivative constraints impose stronger requirements on the differentiability of the underlying functions of time described by qualitative behaviors. However, since HOD constraints are only applied at isolated critical points of the behavior, strictly speaking, such a variable $v(t)$ needs only to be differentiable to the degree necessary to determine the first non-zero derivative, and then only at the isolated point t_i.

Under many circumstances in analysis, for example whenever using Taylor series, one restricts one's attention to *analytic* functions: functions whose higher-order derivatives exist for all orders, over the domain of interest. Fortunately, most of the familiar mathematical functions — polynomials, exponentials, trigonometric functions, etc. — are analytic at all points where they are defined. However, an important fact is that, if a function is analytic over an interval, and is constant over any open sub-interval, it must be constant over the entire interval.

In the examples in this paper, QSIM restricts its attention to analytic functions, by filtering out any behavior which is constant over an interval without being constant everywhere. Thus, the only consistent behaviors for the two-tank cascade (figs. 5, 11) has both tanks reaching their final values simultaneously at $t = \infty$.

However, if we allow non-analytic solutions, we obtain a finite number of additional intuitively reasonable solutions. For example, in the two-tank cascade, we obtain a solution in which the level of water in tank A reaches its final value at finite time, and remains constant while tank B continues to fill (fig. 14).

This prediction corresponds intuitively with real-world observations of processes acting at different time scales: the faster one apparently reaches its limit significantly before the slower one. Two variables may be approaching their limits exponentially and asymptotically, but the more rapidly converging of two exponentials will pass below the level of observability very swiftly, and thereafter appear constant for all practical purposes.

The following table shows the number of predicted behaviors for the

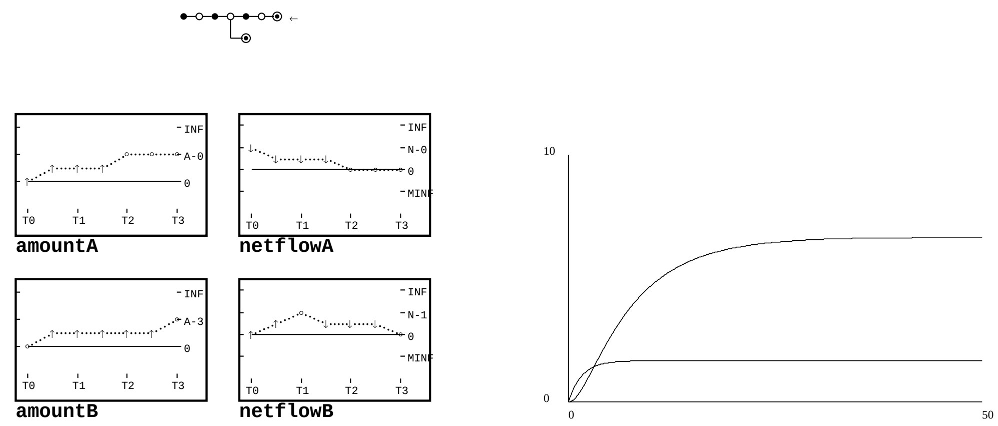


Figure 14: A non-analytic qualitative behavior.
Without the restriction to analytic functions, QSIM predicts a second behavior for the two-tank cascade in which tank A reaches its limit before tank B. While this behavior is not strictly consistent with a linear model, note how closely it resembles the numerically simulated behavior of such a system.

N-tank cascades, with and without the analytic function restriction:

Mechanism	Analytic functions only?		States created
	Yes	No	
two-tank cascade	1	2	9
three-tank cascade	1	8	38
four-tank cascade	1	40	189
five-tank cascade	1	224	1044

While this method provides an indication of the possible time-scale relations in a mechanism, more rigorous methods are available [Kuipers, 1987b] for expressing time-scale abstraction in complex mechanisms.

B Deriving the Curvature Constraint

```
; These rules do the transformations for the curvature constraint.
;   - The first clause in the rule is matched against the sd2 expression.
;   - Additional clauses before "->" are matched against QDE constraints,
;     after substitutions.
;   - The clause after the "->" has bindings substituted, and is returned.

(defparameter *transformation-rules*
  '(((sd2 ?x) (M+ ?x ?y) -> (sd2 ?y))
    ((sd2 ?y) (M+ ?x ?y) -> (sd2 ?x))
                                                    ;
    ((sd2 ?x) (M- ?x ?y) -> (- 0 (sd2 ?y)))
    ((sd2 ?y) (M- ?x ?y) -> (- 0 (sd2 ?x)))
                                                    ;
    ((sd2 ?z) (add ?x ?y ?z) -> (+ (sd2 ?x) (sd2 ?y)))
    ((sd2 ?x) (add ?x ?y ?z) -> (- (sd2 ?z) (sd2 ?y)))
    ((sd2 ?y) (add ?x ?y ?z) -> (- (sd2 ?z) (sd2 ?x)))
                                                    ;
    ((sd2 ?z) (mult ?x ?y ?z) -> (+ (* ?y (sd2 ?x))
                                    (+ (* ?x (sd2 ?y))
                                       (* 2 (* (sd1 ?x) (sd1 ?y))))))
    ((sd2 ?x) (mult ?x ?y ?z) -> (- (/ (sd2 ?z) ?y)
                                    (- (* 2 (* (sd1 ?z)
                                               (/ (sd1 ?y) (^ ?y 2))))
                                       (- (* 2 (* ?z (/ (^ (sd1 ?y) 2)
                                                        (^ ?y 3))))
                                          (* ?z (/ (sd2 ?z) (^ ?y 2)))))))
    ((sd2 ?y) (mult ?x ?y ?z) -> (- (/ (sd2 ?z) ?x)
                                    (- (* 2 (* (sd1 ?z)
                                               (/ (sd1 ?x) (^ ?x 2))))
                                       (- (* 2 (* ?z (/ (^ (sd1 ?x) 2)
                                                        (^ ?x 3))))
                                          (* ?z (/ (sd2 ?z) (^ ?x 2)))))))
                                                    ;
    ((sd2 ?x) (minus ?x ?y) -> (- 0 (sd2 ?y)))
    ((sd2 ?y) (minus ?x ?y) -> (- 0 (sd2 ?x)))
                                                    ;
    ((sd2 ?x) (d/dt ?x ?y) -> (sd1 ?y))
    ((sd2 ?x) (independent ?x) -> 0)
    ((sd1 ?x) (chattering-variable ?x) -> 0)
    ))
```

Inspection of these algebraic transformations reveals that the expressions that can be derived for $sd2(var, t)$ have a very restricted form. In BNF:

$$
\begin{array}{rcl}
\langle exp \rangle & ::= & \langle primitive \rangle \mid \langle compound \rangle \\
\langle primitive \rangle & ::= & \langle number \rangle \mid \langle variable \rangle \mid (qdir\ \langle variable \rangle) \\
\langle compound \rangle & ::= & (\langle bin\_op \rangle\ \langle exp \rangle\ \langle exp \rangle) \mid (\hat{}\ \langle exp \rangle\ \langle number \rangle) \\
\langle bin\_op \rangle & ::= & + \mid - \mid * \mid /
\end{array}
$$

In particular, there are no explicit derivative or monotonic function constraints in the expression, and an expression may only be raised to a constant power.