# Commonsense Reasoning about Causality: Deriving Behavior from Structure*

**Benjamin Kuipers**

*Laboratory for Computer Science, Massachusetts Institute of Technology, Cambridge, MA 02139, U.S.A.*

ABSTRACT

*This paper presents a qualitative-reasoning method for predicting the behavior of mechanisms characterized by continuous, time-varying parameters. The structure of a mechanism is described in terms of a set of parameters and the constraints that hold among them: essentially a 'qualitative differential equation'. The qualitative-behavior description consists of a discrete set of time-points, at which the values of the parameters are described in terms of ordinal relations and directions of change. The behavioral description, or envisionment, is derived by two sets of rules: propagation rules which elaborate the description of the current time-point, and prediction rules which determine what is known about the next qualitatively distinct state of the mechanism. A detailed example shows how the envisionment method can detect a previously unsuspected landmark point at which the system is in stable equilibrium.*

## 1. Introduction

People have a fundamental desire to understand how things work, and an equally fundamental desire to explain their understanding to others. In this paper, I describe a class of knowledge structures to support prediction, explanation and question answering using causal descriptions of physical systems. Within the following framework for causal reasoning (inspired by De Kleer [6, 7]), I address the problem of how a qualitative description of the behavior of a system is derived from a qualitative description of its structure.

Structural description → Behavioral description → Functional description

*This research was supported in part by NIH Grant LM 03603 from the National Library of Medicine.

The *structural description* consists of the individual variables that characterize the system and their interactions; it is derived from the components of the physical device and their physical connections. The *behavioral description* (or *envisionment*) describes the potential behaviors of the system as a network of the possible qualitatively distinct states of the system. I reserve the term *functional description* for a description that reveals the purpose of a structural component or connection in producing the behavior of a system. Thus, the *function* of a steam-release valve in a boiler is to prevent an explosion; the *behavior* of the system is simply that the pressure remains below a certain limit. The existing literature frequently obscures this distinction by using the term 'function' to refer to behavior.

The goal of this research is to develop a knowledge representation capable of describing human commonsense reasoning and explanation about physical causality. *Commonsense* causal reasoning is qualitative reasoning about the behavior of a mechanism which can be done without external memory or calculation aids, although it may draw on concepts learned from the advanced study of a particular domain, e.g. automobile mechanics, computer architecture, or medical physiology. In órder to be useful for modeling human commonsense knowledge, the computational primitives of our representation must not require excessive memory or processing resources.

Simulation of the behavior of a mechanism is useful, for example in medical diagnosis, for determining the consequences of a hypothesized primary change, for predicting the expected course of the patient's disease, and for investigating the effects of hypothetical therapies. *Qualitative* simulation is important because the physician typically lacks precise numerical values for many parameters characterizing the patient's state, and some parameters may be difficult or impossible to measure. In spite of this, the physician is clearly capable of making useful predictions. The knowledge representation described here was inspired by the attempt to capture the knowledge revealed by a physician explaining a case of kidney disease with an unusual presentation. A detailed analysis of the physician's behavior is presented elsewhere [20].

In this paper, I propose a simple but very general descriptive language for structural descriptions, and a qualitative-simulation process for producing the behavioral description. The representation described here begins with a description of the structure of a mechanism that is similar to, but weaker than, a differential equation. The qualitative simulation produces a description of its behavior that corresponds to, but is weaker than, the continuous function that is a solution to the differential equation. Thus, the representation is intended to produce a useful qualitative description of behavior, starting with a qualitative description of structure that would be too weak to support more traditional reasoning methods (see Fig. 1).

Within the structural description, a mechanism is described as a collection of *constraints* holding among time-varying, real-valued *parameters*. The behavioral

$$\begin{tikzcd}
\text{Differential equation} \arrow[r, "\text{Numerical or analytic solution}"] \arrow[d] & f:\mathbb{R} \to \mathbb{R} \arrow[d] \\
\text{Structural description} \arrow[r, "\text{Qualitative simulation}"'] & \text{Behavioral description}
\end{tikzcd}$$

FIG. 1. The qualitative structural description is capable of capturing a less complete state of knowledge than a differential equation, and the qualitative simulation produces a partial description of the mechanism's behavior. Because the qualitative simulation uses heuristics, the two paths through the above diagram do not always yield the same result.

description consists of a finite set of *time-points* representing the qualitatively distinct states of the system, and *values* for each parameter at each time-point. A *value* is a description of the real number corresponding to a parameter at a particular time-point. This description consists of the *ordinal relations* holding among the different values in the behavioral description, and the *IQ value* (the sign of the time derivative) of the parameter at that time. The envisionment proceeds first by *propagating* the implications of initial facts through the constraints to complete a description of the system's state at the current time-point, and second by *predicting* the characteristics of the next distinct qualitative-state description.

After reviewing related work, a simple example demonstrates the basic properties of the representation and the envisionment process. Then a more elaborate example shows how, without external information, the simulation process deduces the existence of a previously unsuspected landmark value, and shows that the mechanism moves to a stable equilibrium about that value. The envisionment process has been implemented in MACLISP, as a program called ENV [10], which has run all the examples included here. The figures presenting the results of the envisionment have been laid out by hand for publication and are not in the actual output format. Appendices provide more formal specifications of the representation and the envisionment process, as well as an additional example addressing related issues.

## 2. Related Research

Answering a question about the behavior of a physical system involves two quite different operations. Problem *formulation* includes selecting which of several ways the physical situation should be described to allow a deeper examination. Problem *solving*, in the narrow sense, starts with a formal description of a well-structured problem and derives an acceptable solution. The different approaches to problem formulation taken by experts and novices have been examined by psychologists such as Chi, Feltovich, and Glaser [3] and

Larkin, McDermott, Simon, and Simon [22]. In particular, when solving problems in physics Chi et al. show that experts describe the given situation in terms appropriate to the underlying physical principle (e.g. conservation of momentum) required for a solution, while novices describe the same situation in terms of the physical objects in the surface statement of the problem. Naturally, the expert is then able to proceed directly to the solution, while the novice must search a larger space of alternate solutions. The work described here is a method of solving problems previously formulated, applicable at either level of expertise.

Artificial intelligence methods for qualitative reasoning about mechanisms were first developed by Rieger and Grinberg [25], whose knowledge representation consists of ***events***, ***tendencies***, ***states***, and ***state changes***, related by several different types of *causal links*. Their system produces realistic qualitative simulations of the behavior of mechanisms. However, their representation lacks a strong distinction between the structure and behavior of a mechanism, which we feel is critical to causal reasoning. More generally, their representation is ambiguous about whether its elements refer to the structure, potential behavior, or actual behavior of the mechanism being described. The CASNET program of Weiss, Kulikowski, Amarel and Safir [27] uses causal links to propagate confirmation scores among pathophysiological states describing the progression of glaucoma. However, the program has no knowledge of the relationship between physiological mechanisms and pathophysiological states, and so expresses causal relationships known to its authors, rather than doing causal reasoning itself. McDermott [24] has proposed an ambitious temporal logic for reasoning about events, actions, and plans as well as processes involving continuously varying parameters. In a sense, he has taken Rieger and Grinberg's representation based on states and events, and created a much extended representation on a better logical foundation, that is capable of addressing a larger set of issues. His logic, however, is oriented toward expressing the behavioral description as actions and events of various kinds, so the structural description is stated as conditionalized events, not as a separate type of description. Since McDermott's goal is to establish a logical framework for temporal reasoning, he demonstrates his logic by expressing many small example sentences rather than larger inference scenarios. Thus, he does not present a detailed set of rules and axioms for inferring behavior from structure. The aim of the present paper is to specify such a set of rules for causal inference, and to use only those features of a logic needed to express the rules.

The *envisionment* approach to reasoning about mechanisms based on the relationship between structure and behavior, rather than between actions and events, has been developed by researchers such as De Kleer [6, 7] and Forbus [11, 12]. When the qualitative description of a system's state is not strong enough to specify which of several futures it will actually follow, the envisionment becomes non-deterministic and the behavioral description contains a branch. Much of the research on envisionment processes has studied the use of external sources of information (e.g. quantitative [6] or teleological [7]) to resolve non-determinism in the envisionment.

There is little agreement on the exact role or expressive power of the *functional description*, which shows how the structure and components of the system contribute to its ability to perform its overall function [8, 19]. The functional description of a system should make explicit not only what behaviors are possible for a system, but why. Thus, a functional description must include terms that refer implicitly to changes past the final state of the system (e.g. *stable equilibrium*), or even to states that do not occur in the envisionment (e.g. *the steam-release valve prevents explosions*). The *function* of the steam-release valve, for example, must include a teleological relationship with the design process, in which the valve was added to the structure in order to prevent a certain behavior.

There is significant disagreement, as well, about the exact nature of the envisionment process. The main issue is the means for describing continuously variable parameters. De Kleer [6, 7] describes changing parameters according to the sign of the derivative (the *IQ value*, standing for *incremental qualitative value*), and an algebra for propagating IQ values across addition constraints. Forbus [11] observes that IQ values alone are inadequate for more than incremental-perturbation analysis, and expands the description to include the signs and magnitudes of both the amount and the derivative of a parameter. In practice, his system uses only the ordinal relations among quantities belonging to partially ordered *quantity spaces*, rather than performing arithmetic operations on numerical magnitudes. Hayes [16, 17] initially proposed the concept of a quantity space, but his efforts were directed toward developing an adequate ontology for causality involving liquids, and he did not use the quantity space in a significant way in his examples, remaining agnostic about its properties. Thus, there is a recognized need for a qualitative method for reasoning about quantities without losing the fine distinctions needed in particular applications.

Another important issue, not fully addressed by previous proposals, is the ability of the envisionment to detect previously unsuspected points at which qualitatively significant changes take place. Forbus [11] and Hayes [16] both assume that landmark values indicating qualitatively significant changes are provided as part of the initial description of the situation. De Kleer's "roller coaster" envisionment [6] usually makes the same assumption, although it is able to posit a change taking place within an interval if the roller coaster's behavior is different at the two ends. However, the point at which the change takes place is not then introduced into the envisionment for further qualitative reasoning. Determining where that point is, and what its properties are, is passed off to the quantitative-reasoning component. As we shall see in Section 5, the envisionment process proposed here is able to detect a previously unsuspected point at which qualitatively significant changes occur and determine many of its properties, without going beyond qualitative reasoning.

A number of researchers are developing methods for deriving behavior from structure in digital electronics, for the purpose of circuit verification (Barrow [1]) or fault diagnosis (Davis [4, 5], Genesereth [14]). Because of the discrete nature of the parameter values, digital electronics is a significantly different domain from physical systems characterized by continuous, analog parameters. In particular, although the simulation of the device may be symbolic [1], the precise values of the parameters can be described and used, so the simulation is not, strictly speaking, qualitative. Furthermore, current work has studied the propagation of information to establish a coherent state for the circuit at a single instant in time. These reasoning techniques do not address (yet) the *evolution* of the state of a circuit over time. Finally, as we shall see below, there is a relatively small set of possible constraints that may hold among parameters in an analog system, and relatively few ways that a set of changing parameters can change over time. In digital electronics, on the other hand, the constraints that can hold among parameters, and the way future states can depend on the past, are limited only by the set of available or constructible component types. Thus, many of the important issues in deriving behavior from structure will be different in the two domains.

## 3. Two Other Ways to Reason about Physical Systems

We can develop certain aspects of our qualitative-reasoning method by comparison with other formal reasoning methods using differential equations. Physical scientists reason about physical systems by describing the structure of a system with a differential equation, then determining its behavior by solving the equation, either analytically or by numerical simulation. The solution can then be analyzed to detect previously unsuspected landmark values of the system's parameters where its behavior changes qualitatively: zero-crossings, maximum or minimum values, and inflection points. Perturbation analysis of the system in the neighborhood of such a point can reveal the existence of (e.g.) a stable equilibrium. There are two costs to using such a reasoning method: the computational resources to perform its primitive operations, and an interpretation process to construct a meaningful description from its output.

Consider an example of a simple physical system (Fig. 2) consisting of a closed container of gas (at temperature T) receiving heat from a source (at $T_S$).

A commonsense description of the behavior of this system is "*The temperature of the gas increases until it is equal to the temperature of the source*". Our goal is a causal reasoner which can produce a description of this form from a description of the causally-relevant structure of the system.

A numerical simulation [13] of this system requires a complete description, in that the value of each parameter at each point in time must be given as a

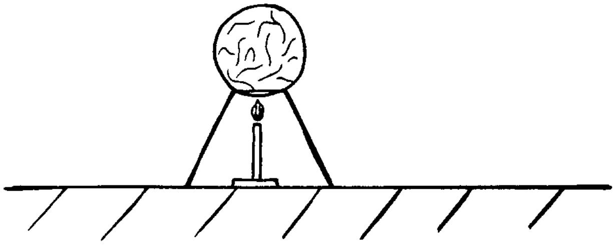


FIG. 2. A container of gas (at temperature T) receiving heat from a source (at a constant temperature $T_s$). There is no heat loss. The rate of flow of heat into the gas (inflow) is a function of the difference ($\Delta T = T_s - T$) between the two temperatures, with $\Delta T = 0$ corresponding to inflow = 0.

real number. The relationship between $\Delta T$ and inflow must also be specified precisely, so we assume strict proportionality with a numerically specified constant. The simulation algorithm is conceptually simple, computing the values of all parameters at a time-point from their values in the previous time-point. It does, however, require arithmetic operations on real numbers, which are more than we might be willing to assume as primitive operations in the human. Fig. 3 gives the structural and behavioral description of the simple heat-flow system, as appropriate for numerical simulation.

The output of the numerical simulation requires substantial further interpretation to recognize and classify important events in the behavior of the system. There is no information about the nature of the dependencies between the different parameters, or how the outcome might vary for different values of the numerical parameters. The fundamental problem is that the numerical description required for this type of simulation has very few states of partial

The structural description:

$$\Delta T = T_s - T$$

$$\text{inflow} = \frac{\Delta T}{10}$$

$$\frac{d}{dt} T = \text{inflow}$$

The behavioral description produced by numerical simulation:

| t | 1 | 2 | 3 | 4 | |
|---|---|---|---|---|---|
| T | 300 | 370 | 433 | 490 | |
| $T_s$ | 1000 | 1000 | 1000 | 1000 | etc. |
| $\Delta T$ | 700 | 630 | 567 | 510 | |
| inflow | 70 | 63 | 57 | 51 | |

FIG. 3. The structural and behavioral description of the simple heat-flow system for numerical simulation. The behavior of the system is described in terms of a discrete set of time-points, each of which specifies the numerical values for the system's parameters.

knowledge: either the value of a parameter is known or it is not. The simulation process cannot run without complete knowledge, and its output can only be matched to a numerically identical system.

The analytic solution of a differential equation provides a substantially different description of the system. In order to describe the causal structure of the simple heat-flow system (Fig. 2) as a differential equation we must specify the relationship between inflow and ΔT explicitly, in this case as a strict proportionality, but with a symbolic constant $k$. It can then be solved analytically as is shown in Fig. 4.

The language of differential equations provides very useful states of partial knowledge about the system, in that quantities may be represented symbolically instead of as real values. There is also a very rich symbolic vocabulary of relationships that may be asserted between quantities in formulating the problem or describing the solution: the arithmetic operators, the trigonometric functions, logarithm, exponentiation, and many others. While these properties make differential equations the fundamental descriptive tool of the physical sciences, they cannot be solved analytically by humans without external memory resources. In spite of this descriptive power, the analytic solution of differential equations requires global and knowledge-intensive operations such as indefinite integration.

We have seen two quite distinct treatments of continuously varying parameters in these two representations. One treats quantities as real numbers, revealing their changes in the course of incremental simulation, but requires a sophisticated interpretation to derive an understanding of the behavior of the mechanism given the simulation. The other representation treats parameters as real-valued continuous functions, and yields an easily interpretable solution, but requires a sophisticated mathematical inference method which often fails to produce a closed-form solution. Qualitative reasoning about physical systems must be able to handle states of incomplete knowledge such as weakly specified functional relationships, and non-numerical initial parameter values. As a form of human commonsense reasoning, it must also require only modest computational facilities [18], but must still be able to handle systems without

$$\frac{\mathrm{d}}{\mathrm{d}t}T = \text{inflow} = k\Delta T = k(T_s - T)$$

$$\int \frac{\mathrm{d}T}{T_s - T} = \int k\,\mathrm{d}t$$

$$\ln(T_s - T) = -kt + C$$

$$T_s - T = C'\mathrm{e}^{-kt}$$

$$T = T_s - C'\mathrm{e}^{-kt}$$

FIG. 4. The first equation is the structural description of the simple heat-flow system (Fig. 2), and the final equation is its behavioral description, created by solving the differential equation analytically.

closed-form solutions, and must be able to recognize unexpected points of qualitative change.

## 4. Qualitative Simulation with Ordinal Quantities

The qualitative simulation, like the other formal models, begins with a structural description which consists of a set of constraints holding among time-varying, real-valued parameters. The three principal types of constraints are:

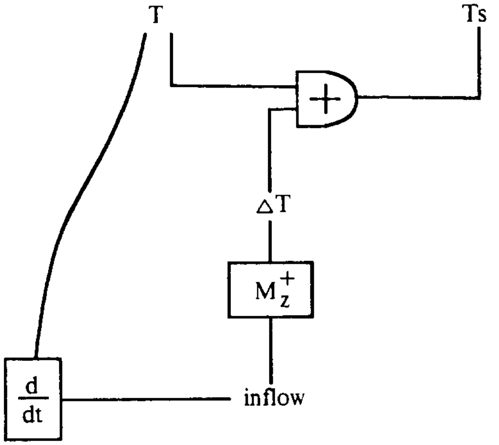
(1) *Arithmetic*: $(X = Y + Z)$. The values of the parameters must have the indicated relationship at each point in time.

(2) *Functional*: $(Y = M^+(X))$. $Y$ is a strictly increasing function of $X$ (decreasing if $M^-$). $M_z^+$ and $M_z^-$ pass through the origin as well.

(3) *Derivative*: $(Y = dX/dt)$. At each time-point, $Y$ is the rate of change of $X$.

The functional relationship, in particular, provides a weaker level of description than is possible with numerical or analytic solutions of differential equations. The relationship $\text{inflow} = M_z^+(\Delta T)$ in Fig. 5 states only that the relationship is strictly monotonically increasing, and that $\text{inflow} = 0$ corresponds to $\Delta T = 0$. Fig. 5 gives the qualitative structure description for the simple heat-flow system in Fig. 2.

The problem we observed in the last section with numerical simulation and analytic solutions of differential equations lies in the restricted states of partial knowledge and in the excessively powerful computational machinery required. In order for qualitative reasoning about physical causality to have more states of partial knowledge with a weaker set of primitive relations, it must operate,



```
(defnet heat-flow
  (unit (temp t-in t-s delta-t)
        (heat inflow)
  (constant (t-s (*range* 0 nil)))
  (constraint
    (adder add A t-in
               B delta-t
               C t-s)
    (m0 + foo X delta-t
              Y inflow)
    (d//dt deriv RATE inflow
                 X t-in))
  (initialize (lt t-in t-s)))
```

FIG. 5. The qualitative causal structure description for the simple heat-flow system (Fig. 2). Note that the $M_z^+$ constraint is a strictly weaker description of the functional relationship than was required for numerical simulation or analytic solution. The defnet form on the right is the actual internal form of the structure description.

not on real numbers, but on symbolic *descriptions* of real numbers and the relations among them. The behavioral description consists of a finite set of *time-points* representing the qualitatively distinct states of the system, and *values* for each parameter at each time-point. A *value* is a description of the real number corresponding to a parameter at a particular time-point. This description consists of the *ordinal relations* (i.e. >, <, and =) holding among the different values in the behavioral description, and the *IQ value* (stated as *increasing*, *steady*, or *decreasing*) of the parameter at that time. Certain values are distinguished or *landmark* values which play a special role in the qualitative simulation. Table 1 summarizes the terminology of this model of qualitative causal reasoning.

Beginning with a set of assertions about the initial state of the system, the envisionment process takes place through the *propagation/prediction cycle.*

*Propagation.* The consequences of information known about the state of the mechanism at the current time-point are propagated through the constraints to create a more complete qualitative description. The current time-point is complete when the direction of change for each value is known. Appendix B provides the detailed specification of the propagation rules.

*Prediction.* The configuration of changing values is examined to determine what can be inferred about the next qualitatively distinct state of the mechanism. A new time-point is defined (or three in case of a branch) and those conclusions asserted within its context. Appendix C provides the detailed specification of the prediction rules.

The prediction rules for determining the next qualitatively distinct state are elaborations on the following three types of qualitative changes, which depend on the ordinal relationship between the current value of a parameter and nearby landmark values.

(1) *Move from landmark value*: If the current value of a changing parameter is equal to a landmark value, then let the next value be perturbed in the direction of change, closer to the starting point than any other landmark value.

(2) *Move to limit*: If the current value of a changing parameter is not equal to a landmark value, and there is a landmark value in the direction of change, let the value of that parameter in the next time-point be equal to the next landmark value.

(3) *Collision*: If there are two changing values moving toward each other, not equal to landmark values nor separated by a landmark value, let their next values be equal, and make that new value a landmark.

Fig. 6 demonstrates these rules graphically.

When the description of the system's current state is not sufficiently complete to determine the next state uniquely, the envisionment branches on the possible states of a particular IQ value or ordinal relation. An additional set of recognition rules (Appendix C) detect properties of the behavioral description, such as cycles, case joins, and quiescence.

Fig. 7 demonstrates that the result of the qualitative simulation is a two-state

TABLE 1. The objects and relations in the qualitative causal reasoning representation. Appendix A provides a more formal definition. Appendix B contains the rules by which individual constraints propagate ordinal and IQ value assertions. The representation for causal knowledge consists of the following objects. They are described here in terms of the real values and real-valued functions for which they provide a partial description.

| Object | Description |
|---|---|
| Parameter | A term corresponding to a continuous real-valued function of time. |
| Switch | A term corresponding to a Boolean-valued function of time. |
| Value | A term corresponding to a real number, the value of a parameter at a particular point in time. |
| Landmark value | A specially designated value. |
| Boolean | A term corresponding to the Boolean value of a switch at a particular point in time. |
| Time-point | A value of the special parameter, *time*. |
| IQ value | A term corresponding to the sign of the derivative of a parameter at a particular point in time. It may have one of three values: increasing (inc), steady (std), or decreasing (dec). |
| Assertion | One of the predicates describing the relation between two values, or between a value and the IQ value at the same time. The reasoning system acquires knowledge about the magnitudes of quantities by inferring new assertions. |
| Ordinal | (⟨rel⟩ ⟨value⟩ ⟨value⟩); ⟨rel⟩ : : = gt \| eql \| lt. |
| IQ | (IQ ⟨value⟩ ⟨iq-value⟩); ⟨iq-value⟩ : : = inc \| std \| dec |
| Constant | (constant ⟨value⟩). |
| Value space | The set of values, partially ordered by the transitive closure of the ordinal assertions. Its primary use is to retrieve the next landmark value in a given direction from a given value. |
| Correspondence | An alist of (parameter landmark-value) pairs consisting of all the parameters at a particular time-point whose values are equal to landmark values. |
| Constraint | One of five types of predicate describing the relationship between several parameters and switches. A set of parameters, switches, and constraints constitutes the structural description of a mechanism, whose behavioral description is determined by examining the assertions generated through qualitative simulation. |
| Arithmetic | (⟨parameter⟩ ⟨parameter⟩ ⟨parameter⟩) [+ *] |
| Functional | (⟨parameter⟩ ⟨parameter⟩) [$M^+$ $M^-$ $M_z^+$ $M_z^-$]. |
| Derivative | (⟨parameter⟩ ⟨parameter⟩ ⟨switch⟩) [d/d*t*]. |
| Inequality | (⟨parameter⟩ ⟨parameter⟩ ⟨switch⟩) [= ≠ < > ⩽ ⩾]. |
| Conditional | (⟨switch⟩ ⟨parameter⟩ ⟨parameter⟩ ⟨parameter⟩). |

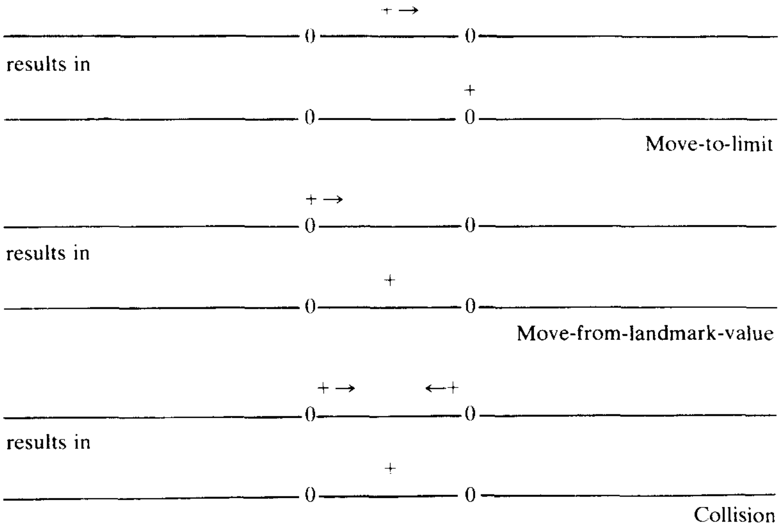


FIG. 6. A graphical illustration of three of the simple prediction rules. The actual set of rules is considerably more complex because there are frequently more than one or two changing values.

envisionment that corresponds closely with the commonsense description of the simple heat-flow system: "*The temperature of the gas increases until it is equal to the temperature of the source*". The qualitative structural and behavioral descriptions offer simplicity of mechanism, in that the qualitative-simulation process depends on the ability to create and match simple assertions, rather than on arithmetic operations or symbolic integration. They also offer the ability to represent partial knowledge, in that both values and functional relationships are only constrained to lie in qualitatively defined classes. Both simplicity of mechanism and states of partial knowledge are valuable properties of a commonsense knowledge representation.

The next example shows that the qualitative simulation can also provide an essential property of a description of physical causality: the ability to detect previously unsuspected values at which qualitatively significant changes take place.

## 5. Detecting and Establishing a Stable Equilibrium

We have seen that the qualitative-simulation process can handle a simple heat-flow problem like the one above, where the system reaches an equilibrium at a previously known landmark value. However, one important product of causal reasoning about a physical system is the existence of previously unsuspected values at which qualitatively significant changes take place. We can explore this issue in the context of a more realistic heat-flow problem, where there are flows of heat both into the gas from the source, and away from the gas into the surrounding cooler air. The causal structure description of the constant($T_s$)

---

(1) $T < T_s$
$\Delta T > 0$
inflow > 0
increasing(T)
decreasing($\Delta$T)
decreasing(inflow)

(2)
$T = T_s$
$\Delta T = 0$
inflow = 0
steady(T)
steady($\Delta$T)
steady(inflow)

---

- Initial conditions are constant ($T_s$) and $T < T_s$.
- In time-point (1), the addition constraint propagates $T < T_s \Rightarrow \Delta T > 0$.
- The functional constraint propagates $\Delta T > 0 \Rightarrow$ inflow > 0.
- The derivative constraint yields inflow > 0 $\Rightarrow$ increasing(T).
- The IQ values propagate similarly to complete the description of time-point (1).
- A version of the move-to-limit rule determines that T, $\Delta$T, and inflow are all changing toward landmark values, and that they must reach their limits simultaneously.
- Time-point (2) is created with the initial assertions $T = T_s$, $\Delta T = 0$, and inflow = 0.
- The constraints propagate ordinal assertions and IQ values until the description of time-point (2) is complete.
- Since all IQ values are steady, the system is quiescent.
- "*The temperature of the gas increases until it is equal to the temperature of the source*".

FIG. 7. The envisionment of Fig. 5 produced by qualitative simulation.

double heat-flow system (Fig. 8) is constructed by merging two descriptions of simple heat flows. The problem is to deduce the existence of an equilibrium temperature ($T_e$) between the temperatures of the heat source ($T_s$) and the air ($T_a$), and to show that the system moves to a stable equilibrium about that temperature.

This description is a qualitative version of the differential equation

$$\frac{\mathrm{d}}{\mathrm{d}t} T = k(T_s - T) - k'(T - T_a)$$

whose solution is

$$T = \frac{kT_s + k'T_a}{k + k'} - C' \mathrm{e}^{-(k+k')t} .$$

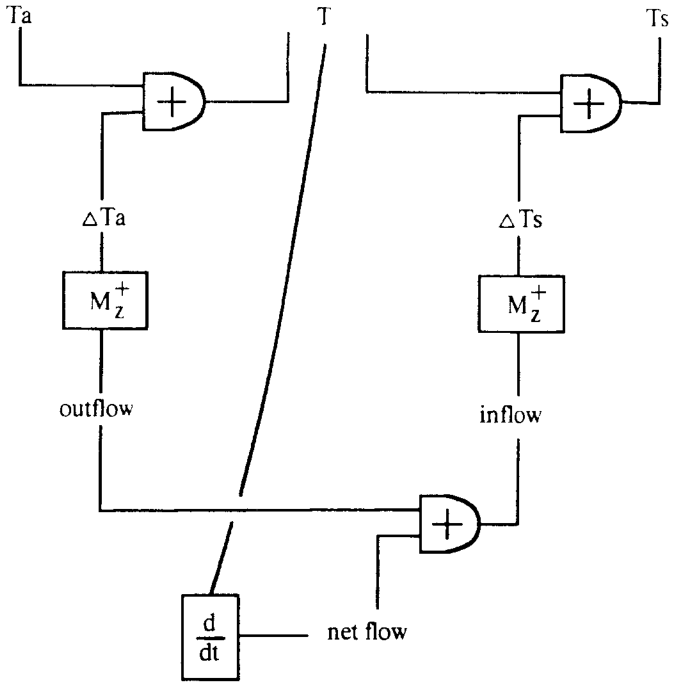


FIG. 8. The causal structure description of the container of gas with two heat flows.

A commonsense description of the behavior of this system is "*The temperature of the gas moves to a temperature between the temperature of the air and that of the source, and remains steady*".

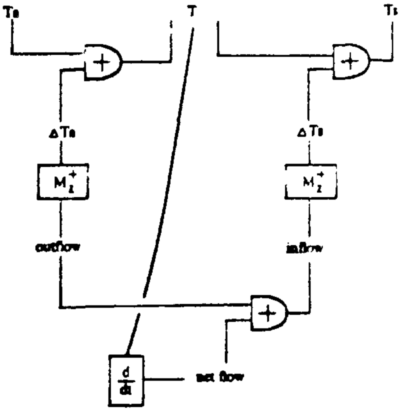
The envisionment process attempts to produce a complete description of the system's behavior through time. As we shall see, it first propagates newly discovered information through the constraints to complete the description of the system at a given time-point. Once that description is sufficiently complete, the envisionment process examines the set of currently changing values to determine the next qualitatively distinct state. If the description of the current state is not sufficiently well specified to determine the next state uniquely, the behavioral description branches according to the three possible states of an unspecified IQ value or ordinal relation. If the alternating cycle of sprouting a new time-point and propagating information among its values should become bogged down in intractible branching, the causal-structure description may be summarized and simplified (cf. Fig. 10 and Appendix D). The new description is less constrained, hence weaker than the old one, but by being simpler it may avoid branching and allow the envisionment process to continue. The envisionment process continues to simulate the system until some terminating condition is detected: quiescence, a cycle, a contradiction, or intractible branching.

The set of values in the behavioral description, partially ordered by the ordinal assertions that are currently known, is called the *value space*. The prediction rules that determine the next state of the system give special status to landmark values. The prediction rules consider only the subset of the value

constant($T_a$)
constant($T_s$)

(1) $T_a < T < T_s$
$\Delta T_a > 0$ $\Delta T_s > 0$
outflow > 0 inflow > 0
net flow = unknown

Case split: relation(net flow, 0)



| (1G) | (1L) | (1E) |
|---|---|---|
| net flow > 0 | net flow < 0 | net flow = 0 |
| inflow > outflow > 0 | 0 < inflow < outflow | inflow = outflow > 0 |
| $T_a < T < T_s$ | $T_a < T < T_s$ | $T_a < T < T_s$ |
| $\Delta T_a$, $\Delta T_s > 0$ | $\Delta T_a$, $\Delta T_s > 0$ | $\Delta T_a$, $\Delta T_s > 0$ |
| increasing(T) | decreasing(T) | steady(T) |
| increasing($\Delta T_a$) | decreasing($\Delta T_a$) | steady($\Delta T_a$) |
| increasing(outflow) | increasing(outflow) | steady(outflow) |
| decreasing($\Delta T_s$) | increasing($\Delta T_s$) | steady($\Delta T_s$) |
| decreasing(inflow) | increasing(inflow) | steady(inflow) |
| decreasing(net flow) | increasing(net flow) | steady(net flow) |

- In time-point (1), starting with the condition that $T_a < T < T_s$, ordinal assertions propagate through the network, producing the succeeding facts, but failing to provide information about net flow.
- In order to allow the derivative constraint to derive IQ values, the envisionment is split into cases according to the sign of net flow. In the branches, with net flow specified, IQ values propagate through the network to complete the description.
- Time-point (1E) is quiescent, with all IQ values steady, so new landmark values are created, and the correspondence between parameters taking on landmark values is recorded.
  (net flow: 0) $\Leftrightarrow$ (inflow: flow*) $\Leftrightarrow$ (outflow: flow*)
  $\Leftrightarrow$ ($\Delta T_a$: $\Delta T_a^*$) $\Leftrightarrow$ ($\Delta T_s$: $\Delta T_s^*$) $\Leftrightarrow$ (T: $T_e$)
- Time-points (1G) and (1L) each contain *six* changing values. However, not enough is known to show that they arrive at their limits simultaneously, making the required case split intractibly large, so the envisionment halts.

FIG. 9. Envisionment of the double heat-flow system. The envisionment diagrams (Figs. 9 and 11) are read from top to bottom, each line following from those above. Each cell contains assertions relevant to a single time-point. Time progresses from top to bottom, and alternate branches are side by side.

space consisting of the current values plus the landmark values. Initially, zero is the only landmark value; the current value of a parameter becomes a landmark when that value has an IQ value of steady.

Figs. 9, 10 and 11 show the stages of the qualitative simulation as it creates the envisionment. Fig. 9 shows how the envisionment of the double-flow system branches in order to derive missing IQ values, how a new landmark point is discovered on one of the branches, and how a set of corresponding values is recorded when several parameters take on landmark values simultaneously. Fig. 10 shows how the structural description is summarized when the first envisionment bogs down at an intractible branch, creating a much more manageable structural description which, though containing much less information, is still a valid description of the system. Fig. 11 shows how the summarized structural description, and the newly discovered correspondence, allow the successor time-points on the remaining two branches to be determined uniquely so the envisionment can be completed. Diagnosis of a stable equilibrium takes place using the final envisionment structure, by showing that a perturbation from the final quiescent state places the system into one of the previously described states from which there is a restoring change.

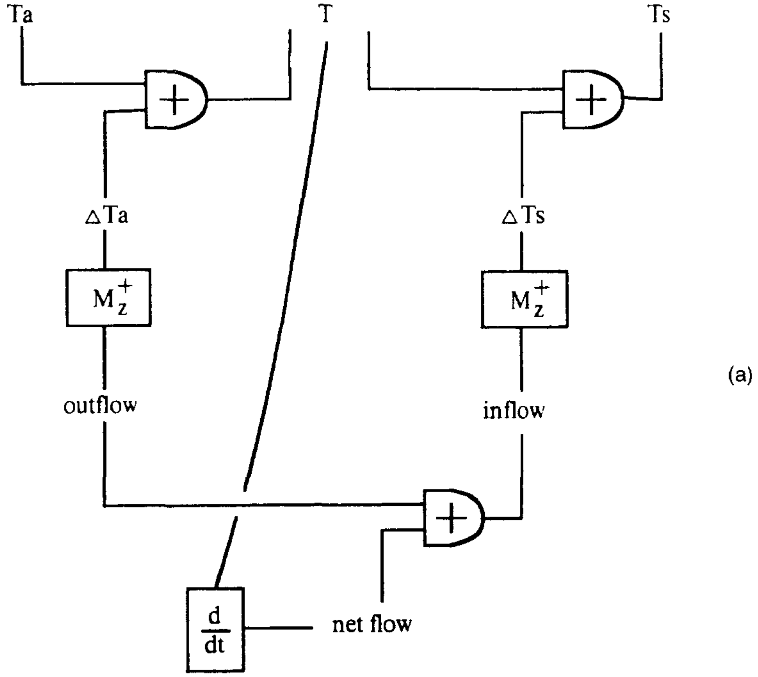


FIG. 10(a).

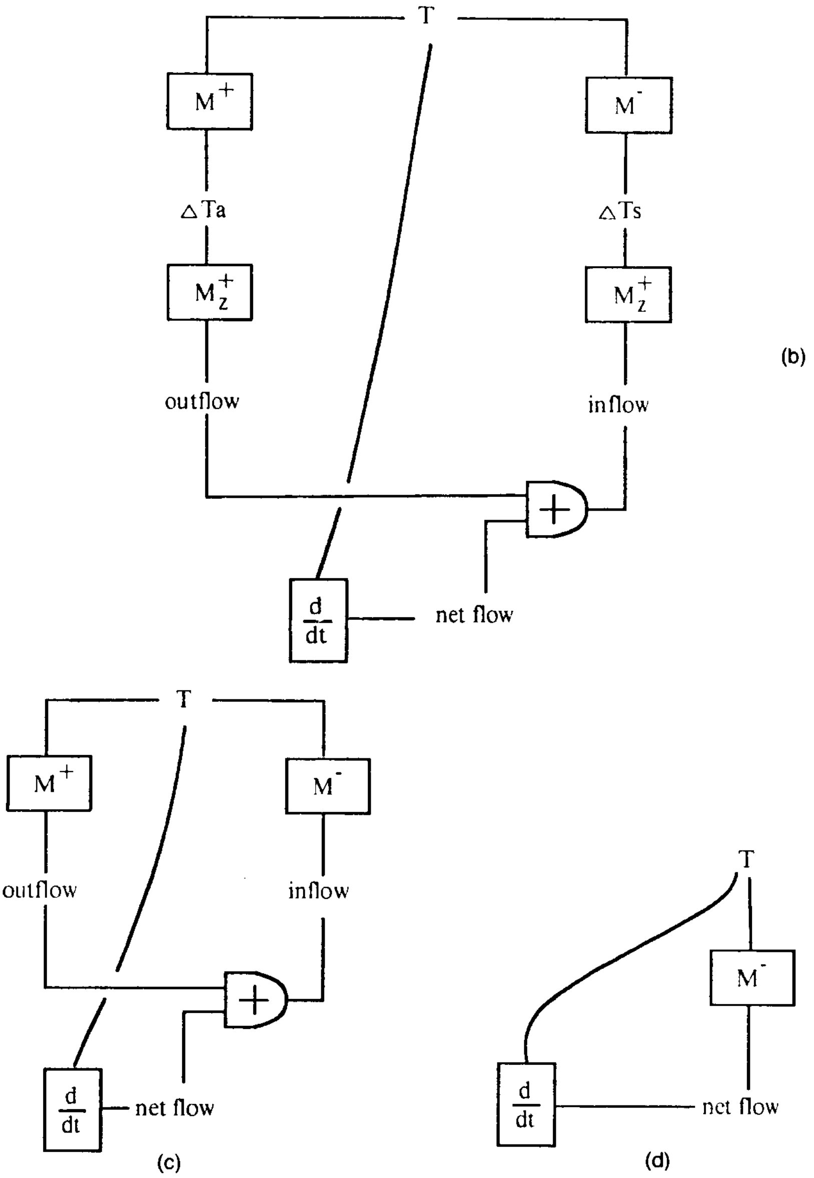


FIG. 10 (b), (c), (d). The arithmetic and functional parts of the causal-structure description are simplified in three steps, applying the following simplification rules. (See Appendix D.) The rules are applied repeatedly until the structural description can not be simplified further.

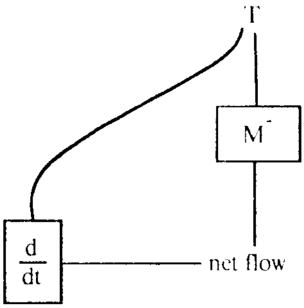
| | |
|---|---|
| $x + y = z$ & constant(y) $\Rightarrow$ $z = M^+(x)$ | (a) → (b) |
| $x + y = z$ & constant(z) $\Rightarrow$ $y = M^-(x)$ | (a) → (b) |
| $y = M^+(M^+(x))$ $\Rightarrow$ $y = M^+(x)$ | (b) → (c) |
| $y = M^-(M^+(x))$ $\Rightarrow$ $y = M^-(x)$ | (b) → (c) |
| $y = M^-(x) - M^+(x)$ $\Rightarrow$ $y = M^-(x)$ | (c) → (d) |

$(T{:}\ T_e) \Leftrightarrow (\text{net flow}{:}\ 0)$

(1)

$T_a < T_e < T_s$
$T_a < T < T_s$
net flow = unknown

Case split: relation(net flow, 0)



| (1G) | (1L) | (1E) |
|---|---|---|
| net flow > 0<br>$T_a < T < T_e$<br>increasing(T)<br>decreasing(net flow) | net flow < 0<br>$T_e < T < T_s$<br>decreasing(T)<br>increasing(net flow) | net flow = 0<br>$T = T_e$<br>steady(T)<br>steady(net flow) |
| (2G)<br>$T = T_e$<br>net flow = 0<br>steady(T)<br>steady(net flow) | (2L)<br>$T = T_e$<br>net flow = 0<br>steady(T)<br>steady(net flow) | (2E)<br>$T = T_e$<br>net flow = 0<br>steady(T)<br>steady(net flow) |

Case join: identical outcomes on all branches

(2)

net flow = 0
$T = T_e$
steady(T)
steady(net flow)

- In time-point (1), ordinal assertions propagate as before, and the need for IQ values prompts a case split.
- Time-point (1E) is quiescent as before.
- The previously determined correspondence makes it possible to infer the relation between T and $T_e$ in time-points (1G) and (1L).
- Since time-points (1G) and (1L) each contain only two changing parameters and their limits are known to correspond, their subsequent states, (2G) and (2L), are easily and unambiguously determined by the move-to-limit rule.
- Since the three branches of the split have identical end-states, they are joined to create state (2). The quiescent state (1E) is copied to an identical but temporally later state (2E) so that the temporal relation between states (1) and (2) is well defined.

FIG. 11. Envisionment of the summarized double heat-flow description.

The envisionment structure, or behavioral description, is now complete, since each state with changing values has a well-defined successor. The overall structure of the envisionment is shown in Fig. 12. Since the envisionment structure has only eight states, it is feasible to examine it for global properties such as the nature of its equilibrium.

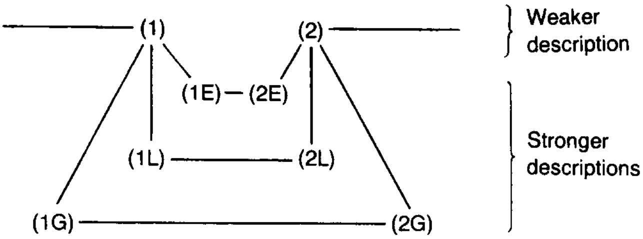


FIG. 12. The qualitative description of behavior is sufficiently compact that it can be examined for global properties such as stable equilibrium.

Since the system ends in a quiescent state, a set of recognition rules is applied to determine whether the quiescence can be diagnosed as some type of equilibrium. Perturbations from state (2) put the system into states (1G) or (1L), from which they return to (2), so the system is in stable equilibrium.

It is worth noting that almost the same conclusion would have been reached if the double heat-flow structure (Fig. 8) had been simplified immediately, without the initial envisionment. The reader may find it instructive to work through the envisionment in Fig. 11 without the correspondence given ahead of time. The more complete description is required, however, to show that $T_a < T_e < T_s$, if it is not initially assumed. Furthermore, there is no reason, before doing the initial simulation, for the envisionment process to transform a stronger description into a weaker (though simpler) one.

## 6. On the Qualitative Description of Time

A mechanism changes continuously with time. Thus, there is no 'next' instant after the current one. The qualitative simulation we use here, however, consists of a discrete set of time-points, and we frequently speak of the prediction phase as predicting the "next qualitatively distinct state" of the system.

Consider the example of a ball thrown into the air with velocity $v_0 > 0$ at time $t = 0$. The ball passes through a continuum of states during its journey up, the down again. However, these states are mapped into five distinct qualitative-state descriptions (see Fig. 13).

Each time-point in the sequence produced by the qualitative simulation corresponds to either a point or an open interval in physical time. In the open-interval case, the physical system clearly continues to change, but within the scope of the same qualitative-state description.

## 7. Individual Variation

Individual variation is an important characteristic of commonsense knowledge (cf. [23]). An individual might have the structural description shown in Fig. 14 for the single heat-flow system. The qualitative simulation is similar to that in Fig. 7, in that it matches the commonsense description: "*The temperature of the*

The quantitative-structure description:

$$\frac{d}{dt}y = v, \quad \frac{d}{dt}v = a = -32\,\text{ft/sec}^2 .$$

The quantitative-behavior description:

$$\begin{aligned} &v(0) = 32\,\text{ft/sec} \Rightarrow y(t) = -16t^2 + 32t\,\text{ft} .\\ &y(0) = 0\,\text{ft}. \end{aligned}$$

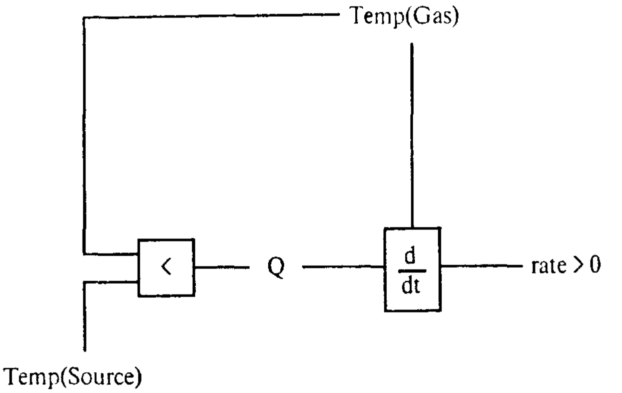
The qualitative-structure description:

$$\frac{d}{dt}y = v, \quad \frac{d}{dt}v = a < 0.$$

The qualitative-behavior description:

| | (0) | (1) | (2) | (3) | (4) |
|---|---|---|---|---|---|
| Y | $Y_0 = 0$ | $0 < Y_1 < YMAX$ | $Y_2 = YMAX$ | $0 < Y_3 < YMAX$ | $Y_4 = 0$ |
| V | $V_0 > 0$ | $0 < V_1 < V_0$ | $V_2 = 0$ | $V_3 < 0$ | $V_4 < V_3 < 0$ |
| A | $A_0 < 0$ | $A_1 < 0$ | $A_2 < 0$ | $A_3 < 0$ | $A_4 < 0$ |

FIG. 13. The structural and behavioral descriptions of the ball system, described both quantitatively and qualitatively. For all t in the open interval $0 < t < 1$, the quantitative descriptions are mapped into the same qualitative description (State (1)). Thus state (1) is the next qualitatively distinct state description after state (0), even though there is clearly no 'next' value of t after $t = 0$.



rate > 0

(1)
$T < T_s$
Q = true
increasing(T)

(2)
$T = T_s$
Q = false
steady(T)

FIG. 14. An alternate causal-structure description of the single heat-flow system, and the corresponding behavioral description. Although the behavioral description is effectively the same as that in Fig. 7 for the simple heat-flow system, the structure description does not generalize to handle the double heat-flow situation.

*gas increases until it is equal to the temperature of the source*". The structural description is correct, but is at the wrong level of detail to support productive causal reasoning because it does not include the fact that the difference between the temperatures of the gas and the source controls the rate of heat increase. This structural description could not be generalized to produce a reasonable envisionment of the double heat-flow example.

We may speculate that some physics students learn models of physical phenomena such as shown in Fig. 14 which are accurately predictive for a certain class of simple mechanisms, but lead to intractible or incoherent structural descriptions when generalized. The Repair Theory approach of Brown and Van Lehn [2], applied to the composition of simple mechanism descriptions to produce more complex ones, may illuminate the misconceptions of naive physics students [15].

## 8. Conclusion

This paper is concerned with the qualitative simulation of physical systems whose descriptions are stated in terms of continuously varying parameters. These continuous systems are interesting because they pose unsolved problems in the representation of knowledge, and because they appear fundamental to commonsense knowledge of causality in the physical world. There appears to be a 'cluster' of knowledge of manageable size about the possible interactions among continuously changing parameters which we can hope to capture and represent [16].

The examples presented above demonstrate a representation for qualitative reasoning about causality in physical mechanisms. The system as described in this paper has been completely implemented in MACLISP. The structural description is essentially a qualitative form of a differential equation, specifying a set of parameters which characterize the state of the mechanism and a set of constraints holding among the parameters. Qualitative simulation produces a behavioral description which specifies the ordinal relationships and directions of change of the parameter values at each point in time.

Just as differential equations do not provide a theory of physics, but rather a language for stating theories of physics, the work presented here is a 'qualitative mathematics' intended as a language for stating theories of qualitative reasoning about particular mechanisms [9]. The preceding discussion of individual variation illustrates this point. Qualitative simulation of behavior from structure is a key element in a complete theory of 'naive physics'. Other critical elements include specifying *which* knowledge to represent to capture the properties of particular domains [17, 21], and specifying how the right structural descriptions can be evoked to handle particular physical situations [11, 12].

Future directions for research on these 'qualitative differential equations' include a mathematical exploration of their properties and the correctness of the qualitative-simulation algorithm (cf. Fig. 1), a reformulation of the prediction rules (Appendix C), and an extension to the formalism to allow time to be treated as a structural, as well as a behavioral, parameter.

## Appendix A. A Formal Definition of the Causal Representation

**Def.** A *parameter* is symbol denoting a continuously differentiable real-valued function of time ($p_i : \mathbb{R} \to \mathbb{R}$).

**Def.** A *constraint* is a pair ⟨P, A⟩ consisting of:

(1) a set P of parameters,

(2) a set A of axioms stating relationships between the values and IQ values of the parameters in P. (See Appendix B.)

**Def.** A *structural description* is a 4-tuple ⟨P, U, C, A⟩ consisting of:

(1) a set P of parameters,

(2) a set U of subsets of P, called *units*, partitioning P into mutually exclusive subsets,

(3) a set C of constraints, holding among the parameters in P,

(4) a set A of axioms stating additional, situation-specific relationships between the values and IQ values of the parameters in P.

(E.g. constant(p) $\Rightarrow$ (for-all (t) (IQ-value(p, t) = steady)).)

**Def.** A *time-point* is a symbol denoting a real number in the domain of some parameter.

**Def.** A *value* is a symbol denoting a real number in the range of some parameter.

**Def.** An *IQ value* is one of the three symbols {increasing, steady, decreasing}, denoting the sign of the derivative of a parameter at a particular time-point.

**Def.** Two landmark values $d_i$ and $d_j$ are *corresponding values* if there is some time-point t and two parameters $p_i$ and $p_j$ related by a monotonic function constraint, such that val($p_i$, t) = $d_i$ and val($p_j$, t) = $d_j$.

**Def.** An *envisionment* is a 7-tuple ⟨SD, T, V, D, R, IQ-value, Corr⟩ consisting of:

(1) a structural description SD,

(2) a set T of time-points, with a subset T* designated as *active* time-points,

(3) a set V of values, with a mapping val : $P \times T \to V$ which is a 1-1 correspondence,

(4) a subset D of V called the *landmark values*,

(5) an order relation R on the elements of V which is a total order when restricted to

the landmark values D in a given unit $U_i$ plus any other value corresponding to a parameter in $U_i$.

(6) a partial mapping $IQ : P \times T \rightarrow$ {increasing, steady, decreasing}, which assigns to each parameter at each time the sign of the derivative of its parameter at that time,

(7) a set Corr of subsets of D denoting corresponding values.

**Def.** An *envisionment process* is a sequence $E_0 \ldots E_n$ of envisionments, where

(1) $E_0$ consists of a structural description SD, and the sets T, V, D, R, IQ, and Corr contain a description of a single, active, initial time-point $t_0$,

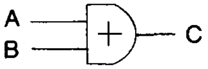
(2) $E_{k+1}$ is derived from $E_k$ by selecting an active time-point from $E_k$, and applying the rules in Appendix C below to that time-point to determine 1 (or 3 in the case of a branch) successor time-points which are added to T, along with a corresponding set of values to V, and possibly additional information to D, R, IQ-value, and Corr,

(3) none of the rules below apply to the final state $E_n$.

## Appendix B. Definition of the Constraints

Constraints are relationships among parameters, but assert ordinal relations and IQ values among the values associated with those parameters at a given time, and also among the values associated with a single parameter at different times. The rules by which these assertions are created are given below. The constraint propagation mechanism is inspired by the scheme developed by Steele [26], modified to propagate ordinal and IQ value assertions rather than integers.

The *addition constraint*:



Ordinal relations can propagate among the values of the adder pins at any given time:

$$A = 0 \Leftrightarrow B = C$$
$$B = 0 \Leftrightarrow A = C$$
$$A > 0 \Leftrightarrow B < C$$
$$A < 0 \Leftrightarrow B > C$$
$$B > 0 \Leftrightarrow A < C$$
$$B < 0 \Leftrightarrow A > C$$

The sign of the derivative of A, B, and C at a given time can propagate through the adder.

$C = A + B$

| + IQ(A)<br>IQ(B) | inc | std | dec |
|---|---|---|---|
| inc | inc | inc | ? |
| std | inc | std | dec |
| dec | ? | dec | dec |

$B = C - A$

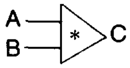
| − IQ(C)<br>IQ(A) | inc | std | dec |
|---|---|---|---|
| inc | ? | inc | inc |
| std | dec | std | inc |
| dec | dec | dec | ? |

When inequalities are derived between values taken on by adder pins at different times, they can be propagated through the adder as well.

$A1 = A2 \;\&\; B1 = B2 \Rightarrow C1 = C2$
$A1 = A2 \;\&\; B1 > B2 \Rightarrow C1 > C2$
$A1 = A2 \;\&\; B1 < B2 \Rightarrow C1 < C2$
$A1 > A2 \;\&\; B1 = B2 \Rightarrow C1 > C2$
$A1 > A2 \;\&\; B1 > B2 \Rightarrow C1 > C2$
$A1 < A2 \;\&\; B1 = B2 \Rightarrow C1 < C2$
$A1 < A2 \;\&\; B1 < B2 \Rightarrow C1 < C2$

$A1 = A2 \;\&\; C1 = C2 \Rightarrow B1 = B2$
$A1 = A2 \;\&\; C1 > C2 \Rightarrow B1 > B2$
$A1 = A2 \;\&\; C1 < C2 \Rightarrow B1 < B2$
$A1 > A2 \;\&\; C1 = C2 \Rightarrow B1 < B2$
$A1 > A2 \;\&\; C1 < C2 \Rightarrow B1 < B2$
$A1 < A2 \;\&\; C1 = C2 \Rightarrow B1 > B2$
$A1 < A2 \;\&\; C1 > C2 \Rightarrow B1 > B2$

The *multiplication constraint*:



Ordinal relations can propagate among the values of the multiplier pins at any given time:

$A = 0 \Rightarrow C = 0$
$B = 0 \Rightarrow C = 0$

$A > 0 \;\&\; B > 0 \Rightarrow C > 0$
$A < 0 \;\&\; B < 0 \Rightarrow C > 0$
$A > 0 \;\&\; B < 0 \Rightarrow C < 0$
$A < 0 \;\&\; B > 0 \Rightarrow C < 0$

$A > 0 \;\&\; C > 0 \Rightarrow B > 0$
$A < 0 \;\&\; C < 0 \Rightarrow B > 0$
$A > 0 \;\&\; C < 0 \Rightarrow B < 0$
$A < 0 \;\&\; C > 0 \Rightarrow B < 0$

[The following rules are only valid assuming A, B, $C > 0$. However, the only examples of multiplication so far are the calculations of concentration and pressure, which all involve physically positive values.]

The sign of the derivative of A, B, and C at a given time can propagate through the multiplier.

$C = A*B$

| $*$ IQ(A) | inc | std | dec |
|---|---|---|---|
| IQ(B) | | | |
| inc | inc | inc | ? |
| std | inc | std | dec |
| dec | ? | dec | dec |

$B = C/A$

| / IQ(C) | inc | std | dec |
|---|---|---|---|
| IQ(A) | | | |
| inc | ? | inc | inc |
| std | dec | std | inc |
| dec | dec | dec | ? |

When inequalities are derived between values taken on by multiplier pins at different times, they can be propagated through the multiplier as well.

$A1 = A2 \;\&\; B1 = B2 \Rightarrow C1 = C2$
$A1 = A2 \;\&\; B1 > B2 \Rightarrow C1 > C2$
$A1 = A2 \;\&\; B1 < B2 \Rightarrow C1 < C2$
$A1 > A2 \;\&\; B1 = B2 \Rightarrow C1 > C2$
$A1 > A2 \;\&\; B1 > B2 \Rightarrow C1 > C2$
$A1 < A2 \;\&\; B1 = B2 \Rightarrow C1 < C2$
$A1 < A2 \;\&\; B1 < B2 \Rightarrow C1 < C2$

$A1 = A2 \;\&\; C1 = C2 \Rightarrow B1 = B2$
$A1 = A2 \;\&\; C1 > C2 \Rightarrow B1 > B2$
$A1 = A2 \;\&\; C1 < C2 \Rightarrow B1 < B2$
$A1 > A2 \;\&\; C1 = C2 \Rightarrow B1 < B2$
$A1 > A2 \;\&\; C1 < C2 \Rightarrow B1 < B2$
$A1 < A2 \;\&\; C1 = C2 \Rightarrow B1 > B2$
$A1 < A2 \;\&\; C1 > C2 \Rightarrow B1 > B2$

The *functional relationship constraints* state that the two parameters so linked have a functional relationship which is either monotonically increasing ($M^+$) or monotonically decreasing ($M^-$), and possibly in which zero corresponds to zero (subscript z).

Y is a *strictly monotonically increasing function* of X:

$$X \longrightarrow \boxed{M_z^+} \longrightarrow Y$$

Information about values of X and Y at a given time can only be propagated if the function passes through the origin (subscript z)

$X > 0 \Leftrightarrow Y > 0$
$X = 0 \Leftrightarrow Y = 0$
$X < 0 \Leftrightarrow Y < 0$

The sign of the derivative at a given time can be propagated.

$\mathrm{IQ}(X) = \mathrm{inc} \Leftrightarrow \mathrm{IQ}(Y) = \mathrm{inc}$
$\mathrm{IQ}(X) = \mathrm{std} \Leftrightarrow \mathrm{IQ}(Y) = \mathrm{std}$
$\mathrm{IQ}(X) = \mathrm{dec} \Leftrightarrow \mathrm{IQ}(Y) = \mathrm{dec}$

An inequality between values of one of the pins at two different times can be propagated to the other pin.

$X1 > X2 \Leftrightarrow Y1 > Y2$
$X1 = X2 \Leftrightarrow Y1 = Y2$
$X1 < X2 \Leftrightarrow Y1 < Y2$

Y is a ***strictly monotonically decreasing function*** of X:

$$X \longrightarrow \boxed{M^-} \longrightarrow Y$$

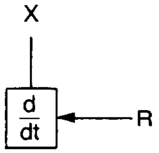
The sign of the derivative at a given time can also be propagated.

$\mathrm{IQ}(X) = \mathrm{inc} \Leftrightarrow \mathrm{IQ}(Y) = \mathrm{dec}$
$\mathrm{IQ}(X) = \mathrm{std} \Leftrightarrow \mathrm{IQ}(Y) = \mathrm{std}$
$\mathrm{IQ}(X) = \mathrm{dec} \Leftrightarrow \mathrm{IQ}(Y) = \mathrm{inc}$

An inequality between values of one of the pins at two different times can be propagated to the other pin.

$X1 > X2 \Leftrightarrow Y1 < Y2$
$X1 = X2 \Leftrightarrow Y1 = Y2$
$X1 < X2 \Leftrightarrow Y1 > Y2$

The *derivative constraint* holds between a parameter and a rate.



At any given time, the sign of the rate can be propagated to the sign of the derivative of X.

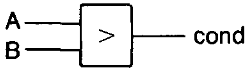
$R > 0 \Leftrightarrow \mathrm{IQ}(X) = \mathrm{inc}$
$R = 0 \Leftrightarrow \mathrm{IQ}(X) = \mathrm{std}$
$R < 0 \Leftrightarrow \mathrm{IQ}(X) = \mathrm{dec}$

There are two versions of these rules, named after the classical theories of motion they resemble [23]. The Aristotelian rule has the benefit of simplicity, while the Newtonian rules are mathematically correct. Appendix E presents an example that illustrates the differences in the behavioral description produced.

Aristotelian:
$\mathrm{IQ}(X1) = \mathrm{std} \Rightarrow X2 = X1$

Newtonian
$\mathrm{IQ}(X1) = \mathrm{std} \ \& \ \mathrm{IQ}(X2) = \mathrm{std} \Rightarrow X2 = X1$
$\mathrm{IQ}(X1) = \mathrm{std} \ \& \ \mathrm{IQ}(X2) = \mathrm{inc} \Rightarrow X2 > X1$
$\mathrm{IQ}(X1) = \mathrm{std} \ \& \ \mathrm{IQ}(X2) = \mathrm{dec} \Rightarrow X2 < X1$

The *inequality constraint* holds between two parameters and a switch, so that the switch holds the boolean value corresponding to the truth of the given relationship between the values of the parameters at that time.

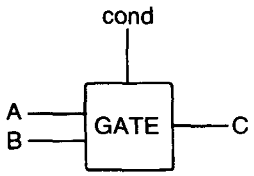


$A > B \Rightarrow \mathrm{cond} = \mathrm{true}$
$A = B \Rightarrow \mathrm{cond} = \mathrm{false}$
$A < B \Rightarrow \mathrm{cond} = \mathrm{false}$

$\mathrm{cond} = \mathrm{true} \Rightarrow A > B$

The *conditional constraint* (gate) holds among three parameters, A, B, and C, and a boolean switch, implementing the relationship

*if cond = true then C = A else C = B*.



$\mathrm{cond} = \mathrm{true} \Rightarrow C = A$
$\mathrm{cond} = \mathrm{false} \Rightarrow C = B$

$C = A \Rightarrow \mathrm{cond} = \mathrm{true}$
$C = B \ \& \ C \neq A \Rightarrow \mathrm{cond} = \mathrm{false}$

## Appendix C. The Envisionment Rules

The *envisionment*, or qualitative simulation of the behavior of a device, proceeds using three types of rules.

(1) *Propagation rules* propagate information across constraints about the values of parameters at a given time-point.

(2) *Prediction rules* determine the nature of the next distinct qualitative-state description from what is known about the current state.

(3) *Recognition rules* detect global properties of the envisionment such as cycles, case-joins, and quiescence.

### C.1. Propagation rules

Propagation rules propagate information about the values of parameters at the current time-point according to the relationships among the parameters and constraints describing the structure of the mechanism.

**Rule P1.** Propagate information (i.e. create a new assertion) for a constraint if enough of its arguments holds new information. (Rules given in Appendix B.)

**Rule P2** (Make landmark value). If a value's IQ-value = steady, make that value a landmark value.

**Rule P3** (Correspondences). If more than one of the values at the current time-point are landmark values, create a *correspondence*: an alist of (parameter landmark-value) pairs consisting of all the parameters whose current values are landmarks, and which are linked directly or indirectly by monotonic function constraints.

**Rule P4** (Contradiction). If propagation derives a contradiction refute the branch containing the value which received the assertion causing the contradiction. If this is the main branch, the entire structural description is at fault.

**Rule P5** (Branch on undetermined rate). If the IQ value of a parameter is unknown at the current time-point, and if it is the 'X' argument of a derivative relation, then branch the envisionment according to the assumptions:

$$\mathrm{IQ}(X) = \mathrm{inc};\ \mathrm{IQ}(X) = \mathrm{std};\ \mathrm{IQ}(X) = \mathrm{dec}.$$

Landmark values are only acquired by being built into the structural description (e.g. the speed limit is 55 mph), or being detected as critical points of functions (Rule P2).

### C.2. Prediction rules

The configuration of changing values (the parameters whose current IQ value is not steady) can be analyzed to select the state or states that immediately succeed the current state. The decision tree below can be seen by inspection to exhaust all cases.

The notation specifies only the values of those parameters which are changing; all others are assumed steady. The current value of a parameter is given by a capital letter, followed by its IQ value (direction of change) in parentheses: A(inc) or B(dec). The value of the same parameter in the time-point created by the envisionment rule is A′ or B′ respectively. Landmark values are starred; sharing the same letter (A and A*) simply signifies that A* is a landmark value which does or will have some important relationship with A. If the result of an envisionment rule is a branch, the rule is written with multiple arrows ('$\Rightarrow$') and consequents.

0. No changing values $\Rightarrow$ no next state.

1. One changing value (A)

1.1 equal to a landmark value: (move from landmark value)

$[A(\text{inc}) = A^*] \Rightarrow [A^* < A']$

1.2 moving toward a landmark value: (move to limit)

$[A(\text{inc}) < A^*] \Rightarrow [A^* = A']$

1.3 not moving toward a landmark value:

$[A(\text{inc})] \Rightarrow [A < A']$ (next state will have same description as current state)

2. Two changing values (A and B)

2.1 both equal to landmark values:

$[A(\text{inc}) = A^*; B(\text{inc}) = B^*] \Rightarrow [A^* < A'; B^* < B']$

2.2 one equal to landmark value:

$[A(\text{inc}) = A^*; B(\text{inc})] \Rightarrow [A^* < A; B < B']$

2.3 neither equal to landmark values:

2.3.1 A and B in different units: not comparable.

2.3.1.1 neither approaching a limit:

$[A(\text{inc}); B(\text{inc})] \Rightarrow [A < A'; B < B']$

2.3.1.2 one approaching a limit:

$[A(\text{inc}) < A^*; B(\text{inc}) \Rightarrow [A^* = A'; B < B']$

2.3.1.3 both approaching limits: (non-deterministic move-to-limit)

$[A(\text{inc}) < A^*; B(\text{inc}) < B^*] \Rightarrow [A^* = A'; B^* = B']$
$\Rightarrow [A^* = A'; B < B' < B^*]$
$\Rightarrow [A < A' < A^*; B^* = B']$

2.3.2 $A < B$

2.3.2.1 both moving the same way: $A(\text{inc}) < B(\text{inc})$

(a) no limits:

$[A(\text{inc}) < B(\text{inc})] \Rightarrow [A' = B']$
$\Rightarrow [A' < B']$

(b) one limit for both values:

$[A(\text{inc}) < B(\text{inc}) < L^*] \Rightarrow [A' = B' = L^*]$
$\Rightarrow [A' < B' = L^*]$
$\Rightarrow [B < A' = B' < L^*]$

(c) one limit point between A and B: (= case 2.3.1.2)

$[A(\mathrm{inc}) < A^* < B(\mathrm{inc})] \Rightarrow [A' = A^* < B < B']$

(d) two separate limit points: (= case 2.3.1.3)

$[A(\mathrm{inc}) < A^* < B(\mathrm{inc}) < B^*] \Rightarrow [A^* = A' < B^* = B']$

$\Rightarrow [A^* = A' < B < B' < B^*]$

$\Rightarrow [A < A' < A^* < B^* = B']$

2.3.2.2 moving toward each other: $A(\mathrm{inc}) < B(\mathrm{dec})$

(a) no limits between them:

$[A(\mathrm{inc}) < B(\mathrm{dec})] \Rightarrow [A' = B']$

(b) one limit point between them: (= case 2.3.1.3)

$[A(\mathrm{inc}) < L < B(\mathrm{dec})] \Rightarrow [A' = L^* = B']$

$\Rightarrow [A' = L^* < B']$

$\Rightarrow [A' < L^* = B']$

(c) two limit points between them: (= case 2.3.1.3)

$[A(\mathrm{inc}) < A^* < B^* < B(\mathrm{dec})] \Rightarrow [A' = A^* < B^* = B']$

$\Rightarrow [A' = A^* < B^* < B']$

$\Rightarrow [A' < A^* < B^* = B']$

2.3.2.3 moving away from each other: $A(\mathrm{dec}) < B(\mathrm{inc})$

(a) no limits on either side:

$[A(\mathrm{dec}) < B(\mathrm{inc})] \Rightarrow [A' < A < B < B']$

(b) one limit point: (= case 2.3.1.2)

$[A^* < A(\mathrm{dec}) < B(\mathrm{inc})] \Rightarrow [A^* = A' < B']$

(c) a limit point on each side: (=case 2.3.1.3)

$[A^* < A(\mathrm{dec}) < B(\mathrm{inc}) < B^*] \Rightarrow [A^* = A' < B' = B^*]$

$\Rightarrow [A^* = A' < B' < B^*]$

$\Rightarrow [A^* < A' < B' = B^*]$

2.3.3 $A = B$

2.3.3.1 moving same way:

$[A(\mathrm{inc}) = B(\mathrm{inc})] \Rightarrow [A' = B']$

$\Rightarrow [A' < B']$

$\Rightarrow [A' > B']$

2.3.3.2 moving opposite ways:

$[A(\mathrm{dec}) = B(\mathrm{inc})] \Rightarrow [A' < B']$

2.3.4 comparable but unknown relationship:

(branch on relation; then use cases 2.3.2 and 2.3.3)

3. More than two changing values

3.1 If any changing value is equal to a landmark value
or moving in a direction with no limit point,
then perturb each value in the direction of motion
(see cases 1.1 and 1.3).

3.2 If no changing values are equal to landmark values
and some changing values are moving toward limit points
and a correspondence exists among all those limit points,

then the next values of those changing parameters are equal to their limit points and any changing parameters without limits are perturbed in their direction of change.

3.3 If no changing values are equal to landmark values
and some changing values are moving toward limit points
and the limit points divide into exactly two sets of corresponding values,
then branch according to the non-deterministic move-to-limit rule (case 2.3.1.3) and any changing parameters without limits are perturbed in their direction of change.

3.4 Otherwise the current state is declared "Intractible".

We are currently experimenting with alternate formulations of the prediction rules which may enable us to handle certain cases difficult to express in the decision-tree format. Thus the definition of 'intractible' for the envisionment system is likely to change.

### C.3. Recognition rules

Recognition rules recognize global configurations in the envisionment that allow the set of time-points to be simplified.

**Rule R1.** If all IQ values are steady, then *recognize a quiescent system*. Remove the current time-point from the set of active time-points. If there are no more active time-points, stop.

**Rule R2.** If all values at the current time-point are equal to landmark values, and all values were equal to the same landmark values at a previous time-point, and all IQ values match in the two time-points, then *recognize a cycle*. Replace the current time-point with a pointer to the previous, identical, time-point, and remove the current time-point from the set of active time-points, since its successors are now known.

**Rule R3.** If all values at the current time-point are equal to landmark values, and there is a time-point on an alternate branch all of whose values are equal to the same landmark values, and all of the IQ values match, then *recognize a case-join*. Replace both time-points with pointers to a special case-join descriptor. In case the case-join captures all the surviving branches of a case-split, map the join into a new time-point related to the one at which the case-split occurred. (See Fig. 12 in the text.)

## Appendix D. Summarizing the Structural Description

The syntactic transformation rules for summarizing the causal-structure description are the following (in mathematical notation, rather than graph diagrams). They are applied repeatedly until the structural description cannot be simplified further.

The transformation is implemented by installing the new constraint, linked with the existing parameters. The existing constraints are left in place but 'turned off' so their rules are no longer activated in response to newly asserted values.

1. Arithmetic constraint with one constant.

$$x + y = z \text{ \& constant}(y) \Rightarrow z = M^{+}(x)$$
$$x + y = z \text{ \& constant}(z) \Rightarrow y = M^{-}(x)$$

$$x * y = z \text{ \& } y > 0 \text{ \& constant}(y) \Rightarrow z = M_{z}^{+}(x)$$
$$x * y = z \text{ \& } z > 0 \text{ \& constant}(z) \Rightarrow y = M^{-}(x)$$

2. Composition of functional constraints.

$$y = M^{+}(M^{+}(x)) \Rightarrow y = M^{+}(x)$$
$$y = M^{+}(M^{-}(x)) \Rightarrow y = M^{-}(x)$$
$$y = M^{-}(M^{+}(x)) \Rightarrow y = M^{-}(x)$$
$$y = M^{-}(M^{-}(x)) \Rightarrow y = M^{+}(x)$$

$$y = M_{z}^{+}(M_{z}^{+}(x)) \Rightarrow y = M_{z}^{+}(x)$$
$$y = M_{z}^{+}(M_{z}^{-}(x)) \Rightarrow y = M_{z}^{-}(x)$$
$$y = M_{z}^{-}(M_{z}^{+}(x)) \Rightarrow y = M_{z}^{-}(x)$$
$$y = M_{z}^{-}(M_{z}^{-}(x)) \Rightarrow y = M_{z}^{+}(x)$$

3. Sum of functional constraints with same net effect.

$$y = M^{+}(x) + M^{+}(x) \Rightarrow y = M^{+}(x)$$
$$y = M^{-}(x) + M^{-}(x) \Rightarrow y = M^{-}(x)$$

$$y = M^{+}(x) - M^{-}(x) \Rightarrow y = M^{+}(x)$$
$$y = M^{-}(x) - M^{+}(x) \Rightarrow y = M^{-}(x)$$

$$y = M_{z}^{+}(x) + M_{z}^{+}(x) \Rightarrow y = M_{z}^{+}(x)$$
$$y = M_{z}^{-}(x) + M_{z}^{-}(x) \Rightarrow y = M_{z}^{-}(x)$$

$$y = M_{z}^{+}(x) - M_{z}^{-}(x) \Rightarrow y = M_{z}^{+}(x)$$
$$y = M_{z}^{-}(x) - M_{z}^{+}(x) \Rightarrow y = M_{z}^{-}(x)$$

## Appendix E. Momentum and Cyclic Behavior

In examining the envisionment of the double heat-flow system, people occasionally ask about momentum: "*What if the value keeps on going rather than stopping at its limit*?" Naturally, this can only occur if the system is sufficiently complex to support that behavior, and if that complexity is reflected in the causal structure description. The example of the oscillating spring demonstrates both momentum and the creation of important new landmark values. This system is governed by the differential equation:

$$\frac{\mathrm{d}^2}{\mathrm{d}t^2}X = M_z^-(X)\,,$$

or, more precisely, by the system of equations:

$$\frac{\mathrm{d}}{\mathrm{d}t}X = V\,, \qquad \frac{\mathrm{d}}{\mathrm{d}t}V = A\,, \quad A = M_z^-(X)\,.$$

In addition to demonstrating cyclic behavior, the oscillating spring (Fig. 15) demonstrates the use of the Aristotelian and Newtonian motion rules associated with the derivative constraint (see Appendix B). In particular, the chart below uses the Aristotelian rule applied to the variable V, in the transition from (3) to (4). This has the curious effect that the IQ value of V changes from steady to increasing while V remains equal to VMIN. Only in the transition from (4) to (5) does the change to the IQ value propagate to cause V > VMIN. Thus, when comparing the behavioral description with the physical world, state (4) is not actually distinct from states (3) and (5), but is a computationally required transitional pseudo-state.

The more complex Newtonian motion rule makes the correct transition from (3) directly to (5). It produces the same behavioral description as shown below, omitting states (4), (7), (10), and (14). One may speculate that the obvious differences in rule complexity is one reason for the observed differences in theories of motion, both across history and in naive subjects [23].

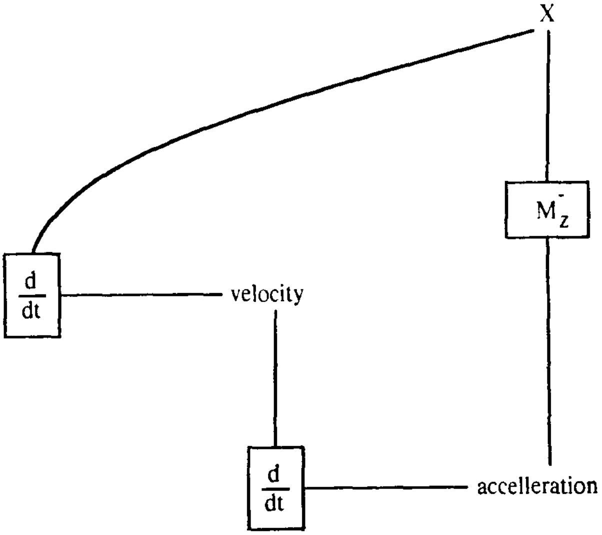


FIG. 15. Causal-structure description: oscillating spring without energy dissipation.

The following chart gives the ordinal assertions and the IQ value assertion for the value of x, v, and a, respectively, at each time-point. The last column gives the rules, other than simple propagation, used to generate each value (Appendix C). The terms that are underlined in each row are the initial information with which that time-point was created from its predecessor.

| | | | | | | | |
|---|---|---|---|---|---|---|---|
| (1) | $\underline{x > 0}$ | std | $\underline{v = 0}$ | dec | $a < 0$ | std | given |
| (2) | $x > 0$ | dec | $\underline{v < 0}$ | dec | $a < 0$ | inc | Rule 1.1 |
| (3) | $\underline{x = 0}$ | dec | $v < 0$ | std | $\underline{a = 0}$ | inc | Rule 3.2 |
| | | | vmin = v3 | | | | Rule P2 (v) |
| (4) | $\underline{x < 0}$ | dec | $v = vmin < 0$ | inc | $\underline{a > 0}$ | inc | Rule 2.1 |
| (5) | $x < 0$ | dec | $\underline{vmin < v < 0}$ | inc | $a > 0$ | inc | Rule 3.1 |
| (6) | $x < 0$ | std | $\underline{v = 0}$ | inc | $a > 0$ | std | Rule 3.2 |
| | xmin = x6 | | | | amax = a6 | | Rule P2 (x, a) |
| (7) | $x = xmin < 0$ | inc | $\underline{v > 0}$ | inc | $a = amax > 0$ | dec | Rule 1.1 |
| (8) | $\underline{xmin < x < 0}$ | inc | $v > 0$ | inc | $\underline{0 < a < amax}$ | dec | Rule 3.1 |
| (9) | $\underline{x = 0}$ | inc | $v > 0$ | std | $\underline{a = 0}$ | dec | Rule 3.2 |
| | | | vmax = v9 | | | | Rule P2 (v) |
| (10) | $\underline{x > 0}$ | inc | $v = vmax > 0$ | dec | $\underline{a < 0}$ | dec | Rule 2.1 |
| (11) | $x > 0$ | inc | $\underline{0 < v < vmax}$ | dec | $a < 0$ | dec | Rule 3.1 |
| (12) | $x > 0$ | std | $\underline{v = 0}$ | dec | $a < 0$ | std | Rule 3.2 |
| | xmax = x12 | | | | amin = a12 | | Rule P2 (x, a) |
| (13) | $x = xmax$ | dec | $\underline{vmin < v < 0}$ | dec | $a = amin$ | inc | Rule 1.1 |
| (14) | $\underline{0 < x < xmax}$ | dec | $vmin < v < 0$ | dec | $\underline{amin < a < 0}$ | inc | Rule 3.1 |
| (15) | $\underline{x = 0}$ | dec | $\underline{v = vmin}$ | std | $\underline{a = 0}$ | inc | Rule 3.2 |
| MATCH detected with state (3). | | | | | | | Rule R2 |

Summarized cycle (landmark values only):

| | | | | | | |
|---|---|---|---|---|---|---|
| (3) | x = 0 | dec | v = vmin | std | a = 0 | inc |
| (6) | x = xmin | std | v = 0 | inc | a = amax | std |
| (9) | x = 0 | inc | v = vmax | std | a = 0 | dec |
| (12) | x = xmax | std | v = 0 | dec | a = amin | std |
| (3) | x = 0 | dec | v = vmin | std | a = 0 | inc |

## ACKNOWLEDGMENT

Christopher Eliot is responsible for the implementation of this system, and has made substantial contributions to its design. Ken Church, Ken Forbus, Ramesh Patil, and Peter Szolovits have also provided helpful comments.

## REFERENCES

1. Barrow, H., Proving the correctness of digital hardware designs, in: *Proceedings National Conference on Artificial Intelligence*, Washington, DC, August, 1983.
2. Brown, J.S. and Van Lehn, K., Repair theory: a generative theory of bugs in procedural skills, *Cognitive Sci.* **4**(4) (1980).

3. Chi, M.T.H., Feltovich, P.J. and Glaser, R., Categorization and representation of physics problems by experts and novices, *Cognitive Sci.* **5** (1981) 121–152.
4. Davis, R., Shrobe, H., Hamscher, W., Wieckert, K., Shirley, M. and Polit, S., Diagnosis based on description of structure and function, in: *Proceedings National Conference on Artificial Intelligence*, Pittsburgh, PA (August, 1982) 137–142.
5. Davis, R., Diagnosis via causal reasoning: Paths of interaction and the locality principle, in: *Proceedings National Conference on Artificial Intelligence*, Washington, DC, August, 1983.
6. De Kleer, J., Multiple representations of knowledge in a mechanics problem-solver, in: *Proceedings Fifth International Joint Conference on Artificial Intelligence*, Cambridge, MA, August, 1977.
7. De Kleer, J., The origin and resolution of ambiguities in causal arguments, in: *Proceedings Sixth International Joint Conference on Artificial Intelligence*, Tokyo, Japan (August, 1979) 197–203.
8. De Kleer, J. and Brown, J.S., Mental models of physical mechanisms and their acquisition, in: J.R. Anderson (Ed.), *Cognitive Skills and Their Acquisition* (Erlbaum, Hillsdale, NJ, 1981).
9. De Kleer, J. and Brown, J.S., The origin, form and logic of qualitative physical laws, in: *Proceedings Eighth International Joint Conference on Artificial Intelligence*, Karlsruhe, West-Germany, August, 1983.
10. Eliot, C. and Kuipers, B., *ENV manual*, Tufts University TMX Memo No. 15, Medford, MA, 1983.
11. Forbus, K.D., Qualitative reasoning about physical processes, in: *Proceedings Seventh International Joint Conference on Artificial Intelligence*, Vancouver, BC, August 1981.
12. Forbus, K.D., Qualitative process theory, *Artificial Intelligence* **24** (1984) this volume.
13. Forrester, J., *Urban Dynamics* (MIT, Cambridge, MA, 1969).
14. Genesereth, M.R., Diagnosis using hierarchical design models, in: *Proceedings National Conference on Artificial Intelligence*, Pittsburgh, PA, August, 1982.
15. Gentner, D. and Stevens, A. (Eds.), *Mental Models* (Erlbaum, Hillsdale, NJ, 1983).
16. Hayes, P.J., The naive physics manifesto, in: D. Michie (Ed.), *Expert Systems in the Micro Electronic Age* (Edinburgh University Press, Edinburgh, 1979).
17. Hayes, P.J., Naive physics I: Ontology for liquids, Department of Computer Science, University of Essex, 1978.
18. Kuipers, B.J., On representing commonsense knowledge, in: N.V. Findler (Ed.), *Associative Networks: The Representation and Use of Knowledge by Computers* (Academic Press, New York, 1979).
19. Kuipers, B.J., De Kleer and Brown's "Mental Models"; A critique, Tufts University Working Papers in Cognitive Science, No. 17, Medford, MA, 1981.
20. Kuipers, B.J. and Kassirer, J.P., Causal reasoning in medicine: Analysis of a protocol, *Cognitive Sci.* **8** (1984) 363–385.
21. Kuipers, B.J., Programs that understand how the body works, in: *Proceedings Second IEEE Computer Society International Conference and 1983 Stocker Symposium on Medical Computer Science and Computational Medicine* (*MEDCOMP '83*), Athens County, OH, September, 1983.
22. Larkin, J., McDermott, J., Simon, D.P. and Simon, H.A., Expert and novice performance in solving physics problems, *Science* **208** (1980) 1335–1342.
23. McCloskey, M., Caramazza, A. and Green, B., Curvilinear motion in the absence of external forces: Naive beliefs about the motion of objects, *Science* **210** (1980) 1139–1141.
24. McDermott, D., A temporal logic for reasoning about processes and plans, *Cognitive Sci.* **6** (1982) 101–155.
25. Rieger, C. and Grinberg, M., The declarative representation and procedural simulation of causality in physical mechanisms, in: *Proceedings Fifth International Joint Conference on Artificial Intelligence*, Cambridge, MA, August, 1977.
26. Steele, G.L., Jr., The definition and implementation of a computer programming language based on constraints, MIT Artificial Intelligence Laboratory TR-595, Cambridge, MA, 1980.
27. Weiss, S.M., Kulikowski, C.A., Amarel, S. and Safir, A., A model-based method for computer-aided medical decision-making, *Artificial Intelligence* **11** (1970) 145–172.

*Received October 1983*