Qualitative Process Theory

Kenneth D. Forbus*
The Artificial Intelligence Laboratory,
Massachusetts Institute of Technology,
Cambridge, MA 02139, U.S.A.

ABSTRACT
Objects move, collide, flow, bend, heat up, cool down, stretch, compress, and boil. These and other things that cause changes in objects over time are intuitively characterized as processes. *To understand commonsense physical reasoning and make programs that interact with the physical world as well as people do we must understand qualitative reasoning about processes, when they will occur, their effects, and when they will stop. Qualitative process theory defines a simple notion of physical process that appears useful as a language in which to write dynamical theories. Reasoning about processes also motivates a new qualitative representation for quantity in terms of inequalities, called the* quantity space. *This paper describes the basic concepts of qualitative process theory, several different kinds of reasoning that can be performed with them, and discusses its implications for causal reasoning. Several extended examples illustrate the utility of the theory, including figuring out that a boiler can blow up, that an oscillator with friction will eventually stop, and how to say that you can pull with a string, but not push with it.*

1. Introduction

Many kinds of changes occur in physical situations. Objects move, collide, flow, bend, heat up, cool down, stretch, compress, and boil. These and the other things that cause changes in objects over time are intuitively characterized by *processes.* Much of formal physics consists of characterizations of processes by differential equations that describe how the parameters of objects change over time. But the notion of process is richer and more structured than this. We often reach conclusions about physical processes based on very little information. For example, we know that if we heat water in a sealed container the water can eventually boil, and if we continue to do so the container can explode. To understand commonsense physical reasoning we must understand how to reason qualitatively about processes, when they will occur, their effects, and when they will stop. This paper describes *qualitative process theory*, which I have been developing for this purpose.

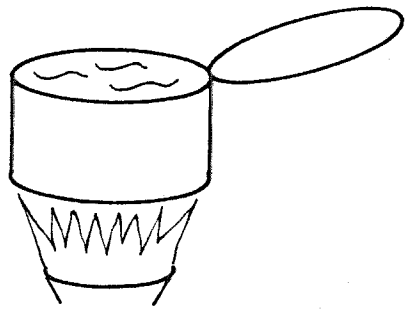
In addition to providing a major part of the representational framework for

* Present address: Department of Computer Science, 1304 West Springfield Avenue, Urbana, IL 61801, U.S.A.

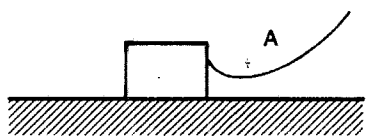
Artificial Intelligence **24** (1984) 85–168
0004-3702/84/$3.00 © 1984, Elsevier Science Publishers B.V. (North-Holland)

commonsense physical reasoning, I expect qualitative process theory to be useful in reasoning about complex physical systems. Programs that explain, repair and operate complex engineered systems such as nuclear power plants and steam machinery will need to draw the kinds of conclusions discussed here. Fig. 1 illustrates some of the commonsense conclusions about physical situations that are discussed in this paper.

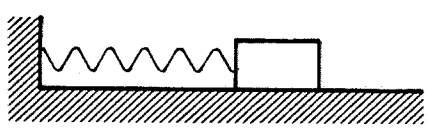
Many schemes have been tried for qualitative reasoning about quantities, including simple symbolic vocabularies (TALL, VERY-TALL, etc.), real numbers, intervals, and fuzzy logic. None are very satisfying. The reason is that none of the above schemes makes distinctions that are relevant to physical reasoning. Reasoning about processes provides a strong constraint on the choice of



Q: What might happen when the heat source is turned on?
A: The water inside might boil, and if the container is sealed it might blow up.



Q: Can we push the block with *A* if it is a string?
A: No, but you can pull the block if it is taut.
Q: Assuming *A* is an elastic band and the block is fixed in position, what might happen if we pull on it?
A: It would stretch and if pulled hard enough would break.



Q: What happens if we release the block?
A: Assuming the spring doesn't collapse, the block will oscillate back and forth. If there is friction it will eventually stop.
Q: What if it gets pumped?
A: If there is no friction the spring will eventually break. If there is friction and the pumping energy is constant then there will be a stable oscillation.

FIG. 1. Some conclusions QP theory can be used to draw.

representation for quantities. Processes usually start and stop when orderings between quantities change (such as unequal temperatures causing a heat flow). In qualitative process theory the value of a number is represented by a *quantity space*, a partial ordering of quantities determined by the domain physics and the analysis being performed. The quantity-space representation appears both useful and natural in modeling a wide range of physical phenomena.

1.1. Motivation

The goal of *naive physics* [21] is to represent the commonsense knowledge people have about the physical world. Here we examine why a theory of processes is needed, what representational burden it carries in naive physics, and the properties such a theory must have.

1.1.1. *Change, histories, and processes*

Reasoning about the physical world requires reasoning about the kinds of changes that occur and their effects. The classic problem which arises is the frame problem [29], namely when something happens, how do we tell what facts change and what facts don't? Using the situational calculus to represent the changing states of the world requires writing explicit axioms that describe what things change and what things remain the same. The number of axioms needed rises as the product of the number of predicates and the number of actions, and so adding a new action potentially requires adding a large number of new axioms. There have been several attempts to fix this problem (e.g., [11, 34], but none of them have seemed adequate. Hayes [21] argues that the situational calculus is fundamentally impoverished, and has developed the notion of *histories* as an alternative.

In situational calculus, situations are used to model the world at different instants in time. Temporally each situation is an instant, but is spatially unbounded. Situations are connected by actions, and actions are specified in terms of what facts can be deduced about the situation which results from performing the action. By contrast, histories are descriptions of objects that are extended through time but are always spatially bounded. Histories are divided into pieces called *episodes*, corresponding to a particular kind of thing happening to the object (episodes will be defined more precisely later on).

Histories help solve the frame problem because objects can interact only when their histories intersect. For example, suppose we are building a clock in our basement. In testing parts of this gadget we look to see what parts touch each other, what parts will touch each other if they move in certain ways, and so on. By doing so we build descriptions of what can happen to the pieces of the clock. We do not usually consider interactions with the furnace sitting in the corner of the basement, because whatever is happening in there is spatially isolated from us (if it is summer it can also be 'temporally isolated').

The assumption that things interact only when they touch in some way also permeates 'non-naive' physics—action at a distance is banished, with fields and particle exchanges introduced to prevent its return. It means that spatial and temporal representations bear most of the burden for detecting interactions. While not easy, developing such representations seems far more productive than trying to develop clever frame axioms. In particular, the qualitative representations of space and time developed in artificial intelligence have precisely the desired properties for reasoning with histories—they often allow ruling out interactions even with very little information.[1]

Histories are to qualitative physical reasoning what descriptions of state parameters over time are to classical numerical simulations. Processes are the analog of the differential equations used to describe the dynamics of the system.

While the classical version of the frame problem is solved, two new problems arise to take its place.

(1) The *local evolution* problem: How are histories generated? Under what circumstances can they be generated for pieces of a situation independently, and then pieced together to describe the whole situation?

In the basement example above, for instance, we could safely ignore the furnace in the corner and concentrate on figuring out how pieces of the clock we are building will move. The divisions are only semi-independent, because certain kinds of changes can violate the conditions for isolation. For example, if the internal thermostat of the furnace gets stuck and it explodes, we can no longer safely ignore it.[2]

(2) The *intersection/interaction* problem: Which intersections of histories actually correspond to interactions between the objects?

Dropping a large steel ball through a flame, for example, won't affect its motion even if the flame is hot enough to melt it unless the gases are moving fast enough to impart significant momentum. Solving these problems in general requires knowing what kinds of things can happen and how they can affect each other—in other words, a theory of processes.

[1]For an example of histories in use, see [12] which describes a program called FROB that reasons about motion through space. FROB used a diagram to compute qualitative spatial representations which served as the spatial framework for its most abstract histories, while the diagram itself was used for the spatial framework in the most exact histories. The description of possible motions it computed was used to assimilate assumptions about the character of the motion (such as assuming a ball would never reach a particular place) and to rule out potential collisions between objects.

[2]Unless the physical situation is simulated by some incremental time scheme, the reasoning involved in extending histories is inherently 'non-monotonic' in the sense of [31]. The reason is that conclusions reached by considering one part of a system may have to be reconsidered in the light of unexpected interactions. In standard incremental time simulations the changes in the entire system are computed over a very short timespan, and then the system is tested to see if any new interactions occur, such as objects colliding. The timespan is usually chosen to be small enough that interactions during a step can be ignored. The cost is that the work required to simulate a system is a function of the time scale rather than the actual complexity of the system's behavior.

In classical mechanics *dynamics* describes how forces bring about changes in physical systems. For any particular domain, such as particles or fluids, a dynamics consists of identifying the kinds of forces that act between the classes of objects in the domain and the events that reslt from these forces. In general, we can view a *qualitative dynamics* as a qualitative theory about the kinds of things that 'can happen' in a physical situation. Qualitative process theory claims that such theories have a common character, in that they are organized around the notion of *physical processes.*

1.1.2. *Reasoning tasks involving qualitative dynamics*

Aside from the role of dynamics in representing change, there are a number of reasoning tasks involving naive physics in which dynamics is central. Each of them is a different 'style' of reasoning, appropriate for solving different classes of problems. The catalog below, while surely incomplete, covers a large proportion of the cases. Examples of inferences from several of these categories are being presented later.

Determining activity: Deducing what is happening in a situation at a particular time. Besides providing direct answers to a class of questions ("what is happening here"?), it is also a basic operation in the other reasoning tasks.

Prediction: Deducing what will happen in the future of some situation. We usually must work with incomplete information, so we can only generate descriptions of possible futures, rather than a single future. De Kleer's notion of *envisioning* is a powerful theory about this type of deduction.[3]

Postdiction: Deducing how a particular state of affairs might have come about. Hayes [22] contains a good example of this kind of deduction. Postdiction is harder than prediction because of the potential necessity of postulating individuals. If we have complete knowledge of a situation and have a complete dynamics, we know what individuals will vanish and appear. But usually there are many ways for any particular situation to have come about. Consider walking back to our basement and finding a small pile of broken glass on the floor. Looking at it we may deduce that a coke bottle was dropped, but we do not know much about its history before that, or about anything else that might have been in the room before we looked. There could have been a troupe of jugglers filling the basement, each manipulating six bottles, and a minor mishap occurred. The simplest explanation is that a single bottle was dropped, but our criteria for simplicity is not due solely to our theories of physics. Postdiction will not be considered further here.

Skeptical analysis: Determining if the description of a physical situation is consistent. An example of this task is evaluating a proposed perpetual motion

[3]Useful as it is, envisioning has certain limitations, especially as a sufficient model of human behavior on this task. See [17] for details.

machine. This kind of reasoning is essential if a reasoner is to recover from inconsistent data and discover inadequacies in its theories about the world.

Measurement interpretation: Given a partial description of the individuals in the situation and some observations of their behavior, inferring what other individuals exist and what else is happening.[4] The first part of a QP-based theory of measurement interpretation is described in [18].

Experiment planning: Given knowledge of what can be observed and what can be manipulated, planning actions that will yield more information about the situation.

Causal reasoning: Computing a description of behavior that attributes changes to particular parts of the situation and particular other changes. Not all physical reasoning is causal, especially as more expert kinds of deductions are considered.[5] Causality seems mainly a tool for assigning credit to hypotheses for observed or postulated behavior. Thus it is quite useful for generating explanations, measurement interpretation, planning experiments, and learning (see [19]).

1.1.3. *Desiderata for qualitative dynamics theories*

There are three properties a theory of dynamics must have if it is to be useful for commonsense physical reasoning. First, a dynamics theory must explicitly *specify direct effects* and *specify the means by which effects are propagated.* Without specifying what can happen and how the things that happen can interact, there is no hope of solving either the local evolution or intersection/interaction problems. Second, the descriptions the theory provides must be *composable.* It should be possible to describe a very complicated situation by describing its parts and how they relate.[6] This property is especially important as we move towards a more complete naive physics that encompasses several 'domains'. In dealing with a single style of reasoning in a particular class of situations an ad hoc domain representation may suffice, but sadly the world does not consist of completely separate domains. Transferring results between several ad hoc representations may be far more complex than developing a useful common form for dynamics theories.[7] Finally, the theory should allow *graceful extension.* First, it should be possible to draw at least the same conclusions with more precise data as can be drawn with weak data. Second, it

[4]Simmons [42] explores the related problem of reconstructing a sequence of events from a static final state, an interesting combination of measurement interpretation and postdiction.

[5]Experts often use arguments based on constraints, such as conservation laws. It seems unlikely that such constraint arguments are central in naive physics, since usually some kind of animistic explanation is proposed to justify them to non-experts (e.g., "the particle senses which path has the least action").

[6]Producing models with this property is a primary motivation for the 'no function in structure' principle [8].

[7]An initial exploration of linking results from reasoning within multiple domains is described in [44].

should be possible to resolve the ambiguities that arise from weak data with more precise information.

These properties are not independent—for example, specifying direct and indirect effects cleanly is necessary to insure composability. Nevertheless, they are not easy to achieve. Graceful extension is bound up with the notion of good qualitative representations. Qualitative representations allow the construction of descriptions that include the possibilities inherent in incomplete information. If designed properly, more precise information can be used to decide between these alternatives as well as perform more sophisticated analyses. Representing quantities by symbols like TALL and VERY-TALL or free space by a uniform grid, for instance, does not allow more precise information to be easily integrated.

It is important to notice that, while qualitative descriptions are approximations, not all approximations are good qualitative descriptions. Changing a value in a qualitative represention should lead to qualitatively distinct behavior. Consider, for example, heating a pan of water on a stove. Suppose we represent the value of the temperature of the water at any time by an interval, and the initial temperature is represented by the interval [70.0, 80.0], indicating that its actual temperature is somewhere between 70 and 80 degrees Fahrenheit. Changing the 'value' of its temperature to [70.0, 85.0] doesn't change our description of what's happening to it (namely, a heat flow), whereas changing it to [70.0, 220.0] changes what we think can be happening to it—it could be boiling as well. While an interval representation makes certain distinctions, they usually are *not* distinctions relevant to physical reasoning.

A purely qualitative theory cannot hope to capture the full scope of human reasoning about physical domains. However, by defining a basic theory using qualitative representations, we can later add theories involving more precise information—perhaps such as intervals—to allow more precise conclusions. In other words, we would like extensions to our basic theory to have the logical character of extension theories—more information should result in a wider class of deductions, not changing details of conclusions previously drawn. In this way we can add theories onto a common base that capture more sophisticated reasoning, such as an engineer uses when estimating circuit parameters or stresses on a bridge.

1.2. Perspective

The present theory has evolved from several strands of work in artificial intelligence. The first strand is the work on *envisioning*, started by De Kleer [6] (see also [7, 12]). Envisioning is a particular style of qualitative reasoning. Situations are modeled by collections of objects with *qualitative states*, and what happens in a situation is determined by running simulation rules on the initial qualitative states and analyzing the results. The weak nature of the information means the result is a directed graph of qualitative states that

corresponds to the set of all possible sequences of events that can occur from the initial qualitative state. This description itself is enough to answer certain simple questions, and more precise information can be used to determine what will actually happen if so desired.

The second strand of work concerns the representation of quantity. Most AI schemes for qualitative reasoning about quantities violate what I call the *relevance principle* of qualitative reasoning—qualitative reasoning about something continuous requires some kind of quantization to form a discrete set of symbols; the distinctions made by the quantization must be relevant to the kind of reasoning being performed. Almost all previous qualitative representations for quantity violate this principle. One exception is the notion of quantity introduced by De Kleer as part of incremental qualitative (IQ) analysis [7], which represented quantities according to how they changed when a system imput was perturbed—increasing, decreasing, constant, or indeterminate. For more general physical reasoning a richer theory of quantity is necessary. IQ analysis alone does not allow the limits of processes to be deduced. For instance, IQ analysis can deduce that the water in a kettle on a lit stove would heat up, but not that it would boil. IQ analysis does not represent rates, so we could not deduce that if the fire on the stove were turned down the water would take longer to boil (Section 5.4 describes how this conclusion might be drawn). The richer notion of quantity provided by QP theory is useful for a wider range of inferences about physical situations than the IQ notion.

The final strand relevant to the theory is, of course, the naive physics enterprise initiated by Pat Hayes [21]. The goal of naive physics is to develop a formalization of our commonsense physical knowledge. From the perspective of naive physics, qualitative process theory is a *cluster*—a collection of knowledge and inference procedures that is sensible to consider as a module. The introduction of explicit processes into the ontology of naive physics should prove quite useful. For instance, in Hayes' axioms for liquids [22] information about processes is encoded in a form very much like the qualitative state idea (see for example axioms 52 through 62). This makes it difficult to reason about what happens in situations where more than one process is occurring at once—Hayes' example is pouring water into a leaky tin can. In fact, difficulties encountered in trying to implement a program based on his axioms for liquids were a primary motivation for developing qualitative process theory.

1.3. Overview of the paper

This paper is an expanded treatment of the central ideas of qualitative process theory [15, 16]. While at this writing certain portions of the theory are still under active development, the ideas described here are fairly stable and other workers have already found these concepts useful. It is hoped that this exposition will stimulate further work in the area.

The next two sections provide the basic definitions for the qualitative

representation of objects, quantities and physical processes. Objects and quantities are discussed first in Section 2 because they are required for defining processes in Section 3. The basic deductions sanctioned by the theory are discussed as well, including analyzing the net effects of processes and the limits of their activity. Section 4 illustrates these deductions by several extended examples, including modeling a boiler, motion, materials, and an oscillator. Further implications of the theory, including causal reasoning, are discussed in Section 5. Section 6 provides a summary, discusses potential applications, and places the theory into the perspective of other recent work in artificial intelligence.

A word on notation. Axioms are used only when they help the reader interested in the fine details. Although a full axiomatic description might be desirable, there are a host of complex technical details involved, few of which essentially contribute to understanding the ideas. When used, axioms are written in a more or less standard sorted predicate calculus notation. The following notational conventions are used for axioms: Predicates and relations are capitalized (e.g., Fluid-Connection), and functions are in lower case (e.g., amount-of, made-of). Sorts are italicized (e.g., *time*). Individuals (often physical objects) are in upper case (e.g., WA) and variables are in lower case (e.g., p). Small finite sets are enclosed by braces ('{','}'). When non-standard notation is introduced an effort is made to show an interpretation of it in terms of logic. This should not necessarily be taken as an endorsement of logic as 'the meaning of' the statements.

At this writing, major parts of the theory have been tested via implementation. The basic deductions sanctioned by the theory (see Section 3.6) are coded, as well as an envisioner for predicting possible behaviors of systems and an algorithm for interpreting measurements taken at an instant. However, the domain models used by the program are still primitive and some of the more sophisticated analyses used in the examples presented here are not yet implemented. In particular, the examples presented should not be taken as representative of the results of a currently running program. This paper does not discuss the implementation at all.

2. Objects and Quantities

To talk about change we first establish some conventions for describing objects and their properties at various times. In this section we describe the temporal notation used and develop the representation of quantity and the *quantity-space* representation for numerical values. *Individual views* are then introduced to describe both the contingent existence of objects and object properties that change drastically with time. The idea of a *qualitative proportionality* ($\propto_Q$) is then introduced to describe functional dependencies between quantities. Finally *histories* are introduced to represent what happens to objects over time.

2.1. Time

We use the representation of time introduced by Allen [1]. To summarize, time is composed of intervals that may be related in several ways, such as one interval being before, after, or equal to another. A novel feature of this representation is that two intervals can *meet*; that is, the start of one interval can be directly after the end of another interval such that no interval lies between them (i.e., time is not dense). Instants are represented as 'very short' intervals which have zero duration but still have distinct beginnings and ends.

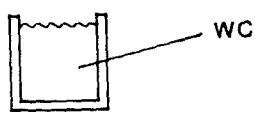
Some additional notation is required. We will assume the functions start and end which map from an interval to the instants that serve as its start or end points. The function during maps from an interval to the set of intervals and instants contained within it. We will assume a function time which maps from instants to some (implicit) global ordering, and a function duration which maps from an interval to a number equal to the difference between the times for the start and the end of the interval. We further assume that the time of the end of a piece of time is never less than the time of its start, so that the duration of an instant is zero while the duration of an interval is greater than zero. Finally, we use the modal operator T to say that a particular statement is true at some time, such as

(T Aligned(PIPE3) I1)

to say that PIPE3 is aligned at (or during) I1. Often the temporal scope of a statement is clear in context, in which case we will not bother with using T.

2.2. Quantities

Processes affect objects in various ways. Many of these effects can be modeled by changing *parameters* of the object, properties whose values are drawn from a continuous range. The representation of a parameter for an object is called a *quantity*. Examples of parameters that can be represented by quantities include



Quantity-Type(amount-of)
Quantity-Type(level)
Quantity-Type(pressure)
Quantity-Type(volume)

Has-Quantity(WC, amount-of)
Has-Quantity(WC, level)
Has-Quantity(WC, pressure)
Has-Quantity(WC, volume)

FIG. 2. Types of quantities. Quantities represent continuous parameters of objects. Here are some quantities that are used in representing the liquid in the cup.

the pressure of a gas inside a container, one-dimensional position, the temperature of some fluid, and the magnitude of the net force on an object.

The predicate Quantity-Type indicates that a symbol is used as a function that maps objects to quantities. To say that an object has a quantity of a particular type we use the relationship Has-Quantity. Fig. 2 illustrates some quantities that pertain to the liquid in a cup.

2.3. Parts of quantities

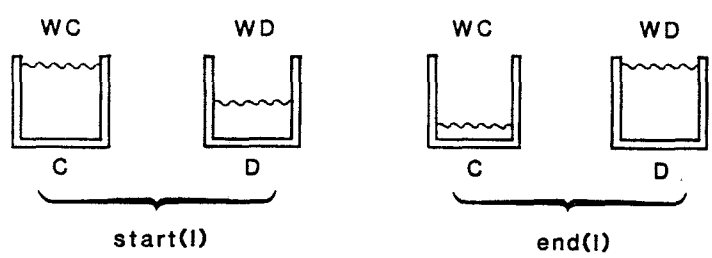
A quantity consists of two parts, an *amount* and a *derivative*. The derivative of a quantity can in turn be the amount of another quantity (for example, the derivative of (one-dimensional) position is the amount of (one-dimensional) velocity). Amounts and derivatives are *numbers*, and the functions A and D map from quantities to amounts and derivatives respectively. Every number has distinguished parts *sign* and *magnitude*. The functions s and m map from numbers to signs and magnitudes respectively. For conciseness, the combinations of these functions that select parts of quantities are noted as:

A_m – magnitude of the amount,
A_s – sign of the amount,
D_m – magnitude of the derivative, or rate,
D_s – sign of the derivative.

Numbers, magnitudes, and signs take on *values* at particular times. When we wish to refer to the value of a number or part of a number, we write

(M Q t).

This statement is read as "the value of Q measured at t". (Notice that M is not the same as m.) Often it is convenient to speak of the value of a quantity, meaning the value of its amount. Fig. 3 illustrates the use of M.



(M A[amount-of(WC)] start(I)) > (M A[amount-of(WD)] start(I))
(M A[amount-of(WC)] end(I)) < (M A[amount-of(WD)] end(I))
(M D_s[amount-of(WC)] I) = −1
(M D_s[amount-of(WD)] I) = 1

FIG. 3. M describes values at different times. Some facts about the two containers expressed as relationships between their quantities.

Signs can take on the values −1, 0, and 1. We take the real numbers as our model for the values of numbers and elements of the non-negative reals as our model for the values of magnitudes so that operations of comparison and combination are well defined.[8] Note however that in basic qualitative process theory we never know numerical values. What we do know about values is described next.

2.4. The quantity space

The value of a number or magnitude is described in terms of its *quantity space.* A quantity space is a collection of numbers which form a partial order. Fig. 4 illustrates a quantity space for the levels of fluid in two tanks C and D connected by a pipe. The orderings and even the elements of a quantity space are not fixed over time. The elements in a particular quantity space are determined by the comparisons needed to establish certain kinds of facts, such as whether or not processes are acting. This means there are only a finite number of elements in any reasonable quantity space, hence there are only a finite number of distinguishable values. Thus the quantity space is a good symbolic description, because it supports case analyses and reasoning by exclusion.

Two elements that are ordered and with no elements in the ordering known to be between them are called *neighbors.* For the quantity space in Fig. 4, level(WD) has height(bottom(D)), height(top(D)), and level(WC) as neighbors, but not height(top(C)). Determining neighbors will be important in determining when processes start and stop acting.

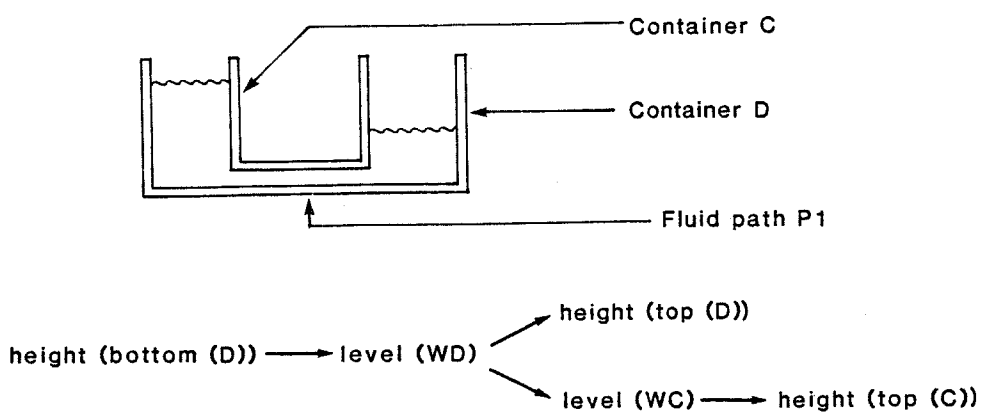


FIG. 4. Graphical notation for a quantity space. WC and WD are the pieces of liquid in containers C and D respectively. The arrow indicates that the quantity at the head is greater than the quantity at the tail. As drawn, level(WC) and height(top(D)) are unordered. For simplicity, we ignore temporal references here.

[8]In this model, m becomes absolute value and s becomes signum, hence the choice of values for signs.

We shall now be a bit more formal about defining quantity spaces and the relationships between parts of quantities. Readers who are uninterested in the details may wish to skip to the next section.

The quantity space of a number consists of a set of elements (numbers or magnitudes, often the amounts of quantities) B and a set of orderings. In basic QP theory the value of a number n is described by the ordering relations between n and the other elements in the quantity space. The value is completely specified only if the orderings among the elements in B is known (i.e., the orderings form a total order), and is incomplete otherwise. Every quantity space can in principle be completely specified. A collection of inequality statements whose union with the orderings of an incompletely specified quantity space results in the quantity space being completely specified is called a *completion* of that quantity space.

All quantity spaces have the distinguished element ZERO. ZERO serves to connect the sign of a number with inequality statements, as follows:

$$\begin{aligned}
&\forall\, n \in \textit{number}\ \ \forall\, t \in \textit{time}\\
&\quad (M\ n\ t) > ZERO \leftrightarrow (M\ s[n]\ t) = 1\\
&\quad \wedge\ (M\ n\ t) = ZERO \leftrightarrow (M\ s[n]\ t) = 0\\
&\quad \wedge\ (M\ n\ t) < ZERO \leftrightarrow (M\ s[n]\ t) = -1
\end{aligned}$$

Note also that the values of magnitudes are related to the values of signs and the value of the number, in that:

$$\begin{aligned}
&\forall\, n \in \textit{number}\, \forall\, t \in \textit{time}\\
&Taxonomy((M\ m[n]\ t) > ZERO, (M\ m[n]\ t) = ZERO)\\
&\wedge\ ((M\ m[n]\ t) = ZERO \leftrightarrow (M\ s[n]\ t) = 0)
\end{aligned}$$

(Taxonomy is drawn from [22] and means that exactly one of its arguments is true.) Thus if the value of D_s for some quantity is 0, then the derivative itself is zero and the quantity is unchanging. We sometimes need to combine sign values across addition. Fig. 5 illustrates the algebra used.

For s[A + B]:

A \ B	−1	0	1
−1	−1	−1	N1
0	−1	0	1
1	N1	1	1

N1: if m[A] > m[B] then s[A]
if m[A] < m[B] then s[B]
if m[A] = m[B] then 0

FIG. 5. Combining sign values. This table specifies how sign values combine across addition. The cases marked by notes require additional information to determine the result.

2.5. Individual views

Objects can be created and destroyed, and their properties can change dramatically. Water can be poured into a cup and then drunk, for example, and a spring can be stretched so far that it breaks. Some of these changes depend on values of quantities—when the amount of a piece of fluid becomes zero we can consider it gone, and when a spring breaks, it does so at a particular length (which may depend on other continuous parameters such as temperature). To model these kinds of changes we use *individual views.*

An individual view consists of four parts. It must contain a list of *individuals*, the objects that must exist before it is applicable. It has *quantity conditions*, statements about inequalities between quantities of the individuals and statements about whether or not certain other individual views (or processes) hold, and *preconditions* that are still further conditions that must be true for the view to hold. Finally, it must have a collection of *relations*, statements that are true whenever the view is true. Fig. 6 illustrates a simple description of the fluid in a cup.

For every collection of objects that satisfies the description of the individuals for a particular type of individual view, there is a *view instance*, or VI, that relates them. Whenever the preconditions and quantity conditions for a VI hold we say that its status is Active, and Inactive otherwise. Whenever a VI is active the specified relations hold between its individuals. An individual view can be thought of as defining a predicate on (or relation between) the individual(s) in the individuals field, and we will often write them that way. The contained liquid description of the previous figure is translated into logical notation in Fig. 7 to illustrate.

```
; we take "amount-of-in" to map from substances and
; containers to quantities

Individual View Contained-Liquid(p)
  Individuals:
    con a container
    sub a liquid
  Preconditions:
    Can-Contain-Substance(con, sub)
  QuantityConditions:
    A[amount-of-in(sub, con)] > ZERO
  Relations:
    There is p ∈ piece-of-stuff
    amount-of(p) = amount-of-in(sub, con)
    made-of(p) = sub
    container(p) = con
```

FIG. 6. Individual views describe objects and states of objects. Here is a simple description of the fluid contained in a cup. This description says that whenever there is a container that contains some liquid substance then there is a piece of that kind of stuff in that container. More elaborate descriptions are developed later on.

```
∀ c ∈ container ∀ s ∈ liquid
  Container(c) ∧ Liquid(s) ⇒
    (∃ IV ∈ view-instance
     ; names of individuals are used as selector functions
     con(IV) = c ∧ sub(IV) = s
     ; logical existence of individual is timeless
     ∧ (∃ p ∈ piece-of-stuff
        container(p) = c ∧ made-of(p) = s)
     ∧ (∀ t ∈ time
        ; it is active whenever Preconditions and Quantity Conditions hold
        (T Status(IV, Active) t)
        ↔ [(T Can-Contain-Substance(con(IV), sub(IV)) t)
           ∧ (T A[amount-of-in(sub(IV), con(IV))] > ZERO t)]
        ; when active, p exists physically and its amount is the
        ; amount of that kind of substance in the container
        ∧ (T Status(IV, Active) t) ⇒
          ((T Contained-Liquid(p) t)
           ∧ Exists-In(p, t)
           ∧ (M amount-of(p) t) = (M amount-of-in(s, c) t))))
; In general,
∀ IV ∈ view-instance ∀ t ∈ time
  (T Taxonomy(Status(IV, Active), Status(IV, Inactive)) t)
```

FIG. 7. The contained liquid description of Fig. 6 translated into logical notation.

The distinction between preconditions and quantity conditions is important. The intuition is to separate changes that can be predicted solely within dynamics (quantity conditions) from those which cannot (preconditions). If we know how a quantity is changing (its D_s-value) and its value (specified as a quantity space), then we can predict how that value will change (Section 3.6). It cannot be predicted within a purely physical theory that someone will walk by a collection of pipes through which fluid is flowing and turn off a valve. Despite their unpredictability, we still want to be able to reason about the effects of such changes when they do occur, hence any dependence on these facts must be explicitly represented. That is the role of preconditions.

2.6. Functional relationships

A key notion of qualitative process theory is that the physical processes and individual views in a situation induce functional dependencies between the parameters of a situation. In other words, by knowing the physics you can tell what, if anything, will happen to one parameter when you vary another. In keeping with the exploration of weak information, we define

$$Q_1 \propto_{Q+} Q_2$$

(read "Q_1 is *qualitatively proportional* to Q_2", or "Q_1 q-prop Q_2") to mean

"there exists a function that determines Q_1, and is increasing monotonic (i.e., strictly increasing) in its dependence on Q_2". In algebraic notation, we would write

$$Q_1 = f(\ldots, Q_2, \ldots).$$

If the function is decreasing monotonic (i.e., strictly decreasing) in its dependence on Q_2, we write

$$Q_1 \propto_{Q-} Q_2$$

and if we don't wish to specify if it is increasing or decreasing,

$$Q_1 \propto_{Q} Q_2$$

For example, we would express the fact that the level of water in a cup increases as the amount of water in the cup increases by adding into the relations of the Contained-Liquid description:

$$\text{level}(p) \propto_{Q+} \text{amount-of}(p).$$

It is important to notice how little information $\propto_Q$ carries. Consider the relationship between level and amount-of stated above. Effectively, all we know is that, barring other changes, when amount-of rises or falls level will also. From this statement alone we do not know what other parameters might affect level, nor do we know the exact way level varies with amount-of. In fact, that $\propto_{Q+}$ statement is satisfied by all of the following equations (assuming appropriate range restrictions):

$$\text{level}(p) = \text{amount-of}(p),$$
$$\text{level}(p) = [\text{amount-of}(p)]^2,$$
$$\text{level}(p) = \sin(\text{amount-of}(p)),$$
$$\text{level}(p) = \text{amount-of}(p) * \text{temperature}(p),$$

and many more.

Often we leave the function implied by $\propto_Q$ implicit. When it is necessary to name the function, we write

Function-Spec(⟨id⟩, ⟨specs⟩)

where ⟨id⟩ is the name of the function being described and ⟨spec⟩ is a set of $\propto_Q$ statements and correspondences (see below) that further specify that function. Suppose for example that level is expressed in a global coordinate system, so

that whenever two open containers whose bottoms are at the same height have fluid at the same level the pressure the fluid exerts (on the bottom, say) is the same. We might introduce a function p-l-fun that relates pressures to levels:

$$\text{Function-Spec(p-l-fun, \{pressure(p)} \propto_{Q+} \text{level(p)\})}.$$

Then if c1 and c2 are containers such that

$$\text{(M level(c1) t0)} = \text{(M level(c2) t0)}$$

then since

$$\text{pressure(c1)} = \text{p-l-fun(level(c1))},$$
$$\text{pressure(c2)} = \text{p-l-fun(level(c2))},$$

by the equalities above we have

$$\text{(M pressure(c1) t0)} = \text{(M pressure(c2) t0)}.$$

Sometimes we want to express the fact that a function depends on something that is not a quantity. In that case we say

$$\text{F-dependency(}\langle\text{id}\rangle, \langle\text{thing}\rangle\text{)}.$$

In the contained-liquid description, for instance, the level depends on the size and shape of the cup as well as the amount of water. Assuming shape and size are functions whose range is something other than quantities, we would write

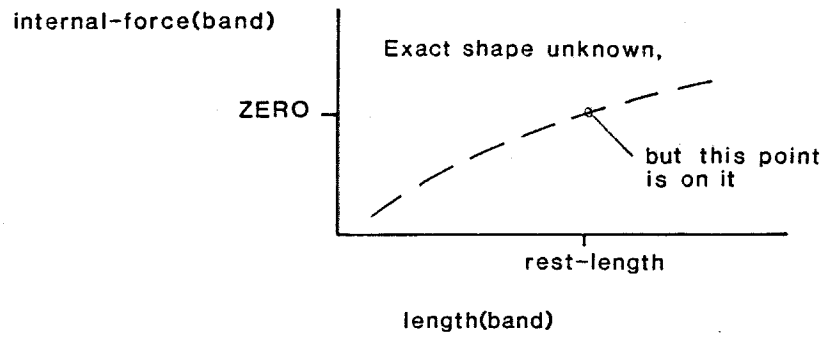
$$\text{Function-Spec(level-function, \{Level(p)} \propto_{Q+} \text{Amount-of(p)\})}$$
$$\text{F-dependency(level-function, Shape(container(p)))}$$
$$\text{F-dependency(level-function, Size(container(p)))}$$

to express this fact. Thus if two containers have the same size and shape, a particular amount of water will result in the same level, but if the size or shape is different we cannot deduce anything about the level of water.

The definition of $\propto_Q$ is motivated in part by issues involved in learning and causal reasoning, and we postpone further discussion of its variants until Section 5. There is one other kind of information that can be specified about the function implied by $\propto_Q$, and that is a finite set of *correspondences* it induces between points in the quantity spaces it connects. An example of a correspondence is that the force exerted by an elastic band B is zero when it is at rest. This would be written:

Correspondence((internal-force(B), ZERO),
(length(B), rest-length(B)))

Correspondences are the means of mapping value information (inequalities) from one quantity space to another via $\propto_Q$. For example, if the length of the band described above is greater than its rest length the internal force is greater than zero. Fig. 8 illustrates.



internal-force(band) $\propto_{Q+}$ length(band)
Correspondence ((internal-force(band), ZERO),
(length(band), rest-length(band)))

FIG. 8. Correspondences link quantity spaces across $\propto_Q$. A correspondence statement allows information about inequalities to be transferred across qualitative proportionalities ($\propto_Q$). The rough shape of the graph is determined by the $\propto_Q$, the equality between the two points is determined by the correspondence.

2.7. Histories

To represent how things change through time we use Hayes' notion of a *history*. We assume the concepts introduced in [22] as our starting point. To summarize, the history of an object is made up of *episodes* and *events*. Episodes and events differ in their temporal aspects. Events always last for an instant, while episodes usually occur over an interval of time. Each episode has a start and an end which are events that serve as its boundaries. Following [1], we assume that episodes and events meet, that is, the start of some piece of history is directly after the end of the previous piece with no time in between. This allows us to say, for example, that the episode of heating water on a stove is ended by event of the water reaching its boiling temperature, yet during the episode the temperature was below the boiling point.

The particular class of histories Hayes introduced are called *parameter* histories, since they are concerned with how a particular parameter of a specific individual changes.[9] Objects can have more than one parameter, and these

[9]In fact, Hayes' examples are parameter histories for 'amount of stuff', representing an object solely as a piece of space-time. The representation introduced here can be thought of as pieces of space-time that are 'bristling with properties'.

parameters often can change independently. For example, if we drop a steel ball past a flame, the ball will heat up a bit but the motion won't be affected (unless the combustion gases impart significant momentum to it). Thus the history of an object includes the union of its parameter histories. Fig. 9 illustrates the parameter histories for the situation just described. The criteria for individuation, for breaking up time into episodes and events (the spatial component of parameter histories is inherited from the object they are a parameter of) are changes in the values of quantities and their parts. In Fig. 9, for example, the events consist of the ball's position reaching h2 and h1, because different values hold before and after that time. The final component of an object's history are the histories for the processes it participates in, which are described in Section 3.7.

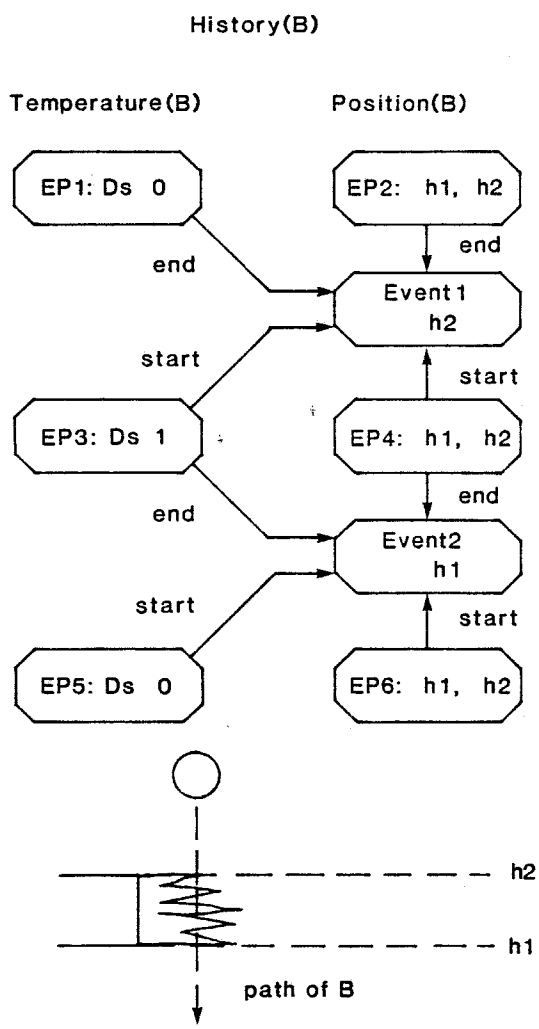


FIG. 9. Parameter histories describe when values change. Part of the parameter histories for a ball being dropped through a flame are depicted below. Time runs from top to bottom, and the portion of the history that depicts what is happening (motion and heat flow) is not shown.

Again following Hayes, a *slice* of a history denotes a piece of an object's history at a particular time. We denote the slice of an individual i at time t by

at(i, t) .

If we let all functions, predicates, and relations that apply to objects apply to slices as well, with functions that map from objects to quantities map from slices to values, then we could be rid of T and M and just talk in terms of slices. For instance, instead of writing

(T Aligned(P1) t0)
(M A[amount-of(WC)] t0) > (M A[amount-of(WB)] t0)

we could write

Aligned(at(P1, t0))
A[amount-of(at(WC, t0))] > A[amount-of(at(WB, t0))]

For clarity of exposition, however, we continue to use T and M.

The notion of history so far is 'object-centered'. Since processes will often act between several objects, we need a way of talking about several objects at a particular time. We recycle the term *situation* to mean a collection of slices for a set of objects under consideration at some particular time. Unlike situational calculus, the temporal aspect of a situation can be either an instant or an interval. Also, a situation is now spatially bounded—its spatial extent is that of the slices that comprise it. In formulae where times are required, we assume a coercion from a situation or event to its time so that we can freely use the names of situations in expressions involving T and M.

The question of what constitutes a useful situation brings us back to the local evolution problem described in the introduction. We may now state it more precisely: Given some collection of objects that we know about at a particular time, can we figure out some way to divide them up into situations that can be considered semi-independently?[10] For the moment we leave the criteria of what constitutes useful situations unspecified, returning to this problem in Section 3.7 after discussing processes.

3. Processes

A physical situation is usually described in terms of a collection of objects, their properties, and the relationships between them. So far our description of the world has been static—we can say that things are different from one time to another, but have not provided the means by which changes actually occur. The ways in which things change are intuitively characterized as processes. A *physical process* is something that acts through time to change the parameters

[10]In current AI systems this problem usually does not arise because the situations under consideration are composed solely of relevant objects. However, as we attempt to make programs that can deal with more realistic problems this issue will become very important.

of objects in a situation. Examples of processes include fluid and heat flow, boiling, motion, stretching and compressing.

This section describes what processes are, including how to specify them, and elaborates the notion of *influences*. A catalog of basic deductions involving processes illustrates the kinds of conclusions that can be drawn within QP theory. Histories are extended to include occurrences of processes, and the role of processes in specifying a language of behavior is discussed.

3.1. Specifying processes

A process is specified by five parts:

(1) the *individuals* it applies to;

(2) a set of *preconditions*, statements about the individuals and their relationships other than quantity conditions;

(3) a set of *quantity conditions*, that are either statements of inequalities between quantities belonging to the individuals (including domain-dependent constants and functions of them) or statements about the status of processes and individual views;

(4) a set of *relations* the process imposes between the parameters of the individuals, along with any new entities that are created;

(5) a set of *influences* imposed by the process on the parameters of the individuals.

Fig. 10 illustrates process specifications for heat flow and boiling. (For fans of logic, Fig. 11 illustrates how the boiling process would look translated into predicate calculus.)

Basically, a process is just like an individual view—it is a time-dependent thing—except it has something called *influences*. To recapitulate, for every collection of objects that satisfy the individuals specification for a particular type of process, there is a *process instance* (PI) that relates them. The process instance is active, representing the process acting between these individuals, exactly whenever both the preconditions and the quantity conditions are true. Preconditions are those factors that are outside QP theory, such as someone opening or closing a valve to establish a fluid path, but still relevant to whether or not a process occurs. The quantity conditions are those statements that can be expressed solely within QP theory, such as requiring the temperature of two bodies to be different for heat flow to occur, or a heat flow to occur as a prerequisite to boiling. The set of relations associated with a process are the relationships it imposes between the objects it is acting on. The relations component usually describes, but is not limited to, indirect effects via functional relationships between quantities, such as the flow rate in fluid flow being qualitatively proportional to the difference in the pressures of the contained fluids involved. The relations also include descriptions of any new individuals created by the process, as for example the steam generated by boiling, and facts needed by external representations, such as describing appearances. We discuss influences next.

```
process heat-flow

Individuals:
  src an object,Has-Quantity(src, heat)
  dst an object,Has-Quantity(dst, heat)
  path a Heat-Path, Heat-Connection(path, src, dst)

Preconditions:
  Heat-Aligned(path)

QuantityConditions:
  A[temperature(src)] > A[temperature(dst)]

Relations:
  Let flow-rate be a quantity
  A[flow-rate] > ZERO
  flow-rate ∝Q+ (temperature(src) − temperature(dst))

Influences:
  I−(heat(src), A[flow-rate])
  I+(heat(dst), A[flow-rate])


process boiling

Individuals:
  w a contained-liquid
  hf a process-instance, process(hf) = heat-flow
                                 ∧ dst(hf) = w

QuantityConditions:
  Status(hf, Active)
  — A[temperature(w)] < A[t-boil(w)]

Relations:
  There is g ∈ piece-of-stuff
  gas(g)
  substance(g) = substance(w)
  temperature(w) = temperature(g)
  Let generation-rate be a quantity
  A[generation-rate] > ZERO
  generation-rate ∝Q+ flow-rate(hf)

Influences:
  I-(heat(w), A[flow-rate(hf)])
  ; The above counteracts the heat flow's influence
  I−(amount-of(w), A[generation-rate])
  I+(amount-of(g), A[generation-rate])
  I−(heat(w), A[generation-rate])
  I+(heat(g), A[generation-rate])
```

FIG. 10. Two examples of process specifications. Heat flow happens between two objects that have heats and are connected via some path through which heat can flow. The predicate Heat-Aligned is true exactly when heat can flow through the path. Boiling happens to a contained liquid being heated, and creates a gas made of the same stuff as the liquid. t-boil represents the boiling point for the piece of stuff involved.

```
∀ w ∈ contained-liquid ∀ hf ∈ process-instance
(process(hf) = heat-flow ∧ dst(hf) = w ⇒
  [∃ pi ∈ process-instance
   process(pi) = boiling ∧ w(pi) = w ∧ hf(pi) = hf
   ∧ [(Status(hf, active) ∧ A[temperature(w)] = A[t-boil(w)])
      ↔ Status(pi, Active)]
   ∧ [Status(pi, Active) ⇒
     [∃ g ∈ piece-of-stuff ∃ generation-rate ∈ quantity
      Boiling(w, hf)
      ∧ gas(g)
      ∧ substance(g) = substance(w)
      ∧ temperature(w) = temparature(g)
      ∧ A[generation-rate] > ZERO
      ∧ generation-rate ∝ Q+ flow-rate(hf)
      ∧ A[flow-rate(hf)] ∈ minus-imputs(influence-adder(heat(w)))
      ∧ A[generation-rate]
        ∈ minus-inputs(influence-adder(amount-of(w)))
      ∧ A[generation-rate]
        ∈ plus-inputs(influence-adder(amount-of(g)))
      ∧ A[generation-rate]
        ∈ minus-inputs(influence-adder(heat(w)))
      ∧ A[generation-rate]
        ∈ plus-inputs(influence-adder(heat(g)))]]]

; a directly influenced quantity has an "influence adder",
; which determines its derivative
∀ q ∈ quantity
  ([¬ minus-inputs(influence-adder(q)) = { }
   ∨ ¬ plus-inputs(influence-adder(q)) = { }] ⇒
     D[q] = Sum(influence-adder(q)))
```

FIG. 11. Boiling expressed as an axiom. How the boiling description could be written as an axiom. For clarity, temporal references have been omitted.

3.2. Influences and integration

Influences specify what can cause a quantity to change. There are two kinds of influences, *direct* and *indirect*. The influences component of a process specifies the direct influences imposed by that process. For example, in a flow process (see Fig. 10) the flow rate will make the amount of 'stuff' at the source tend to decrease and the amount of 'stuff' at the destination tend to increase. If the number n is a direct influence on the quantity Q, we write

$$I+(Q, n) \qquad I-(Q, n) \qquad I\pm(Q, n)$$

according to whether its influence is positive, negative, or unspecified. Importantly, processes are the only source of direct influences. If at least one process is directly influencing a quantity Q at some particular time, then we say that Q is *directly influenced*. If a quantity is directly influenced, its derivative equals the sum of all of the direct influences on it.

An indirect influence occurs when a quantity is a function of some other quantity that is changing. Qualitative proportionalities ($\propto_Q$), introduced earlier, are the means of specifying these effects. Sometimes we will refer to a process or quantity *indirectly influencing* some quantity. One quantity indirectly influences another if the second quantity is qualitatively proportional to the first. A process indirectly influences a quantity Q_1 if it directly influences some quantity Q_2 which in turn indirectly influences Q_1.

Notice that direct influences tell us much more about the relationship between quantities than indirect influences as specified by $\propto_Q$. Given two direct influences on a quantity we know they combine by addition, but we do not know how multiple indirect influences combine because $\propto_Q$ provides little information on the exact form of the underlying function. We will discuss this in more detail in Section 3.6.3.

At any particular time a quantity must be either directly influenced, indirectly influenced, or not influenced at all. Importantly, we assume that no quantity is both directly and indirectly influenced at once. A domain physics that allows a quantity to be both directly and indirectly influenced at the same time is considered to be inconsistent. This may seem odd, given that relationships between quantities in 'real' physics are often specified as constraint equations. For example, we could express the equation $F = m * a$ in three different ways using qualitative proportionalities, each corresponding to one parameter being described as a function of the other two. How, and why, do we select a particular function to represent the constraint relationship?

The choice is made to reflect the way causality works in the domain. In thinking about motion, for instance, we cannot directly apply an acceleration—we can only cause acceleration by imposing a force. Similarly, we cannot by accelerating something or pushing on it cause the mass of a solid object to change, yet its mass will affect how much acceleration we get for a given push. These considerations suggest the proper rendering of $F = m * a$ is:

$$a \propto_{Q+} F \qquad a \propto_{Q-} m .$$

There is a subtle issue lurking here. In a sense, quantities that can be directly influenced are 'independent', in that we can cause changes in them directly via active processes. All other quantities must be changed indirectly as a consequence of the changes processes make on the directly influenced parameters. The choice of directionality when transforming a constraint equation into a function must respect this fact. The full importance of this distinction is discussed later on when examining causal reasoning (Section 5.2).

The influences on a quantity are combined to determine its derivative (we describe just how later). A notion of integrability—the relationship between the derivative of a quantity and its amount—is needed. Essentially, if the derivative is negative then the amount will decrease over an interval, if positive

then the amount will increase, and if zero then the amount will be the same:

$$
\begin{aligned}
&\forall\, q \in \textit{quantity}\ \forall\, I \in \textit{time} \\
&(\text{constant-sign}(D[q], I) \Rightarrow \\
&\quad (M\ D_s[q]\ \text{during}(I)) = -1 \\
&\quad \leftrightarrow (M\ A[q]\ \text{end}(I)) < (M\ A[q]\ \text{start}(I)) \\
&\quad \wedge\ (M\ D_s[q]\ \text{during}(I)) = 1 \\
&\qquad \leftrightarrow (M\ A[q]\ \text{end}(I)) > (M\ A[q]\ \text{start}(I)) \\
&\qquad \wedge\ (M\ D_s[q]\ \text{during}(I)) = 0 \\
&\qquad \leftrightarrow (M\ A[q]\ \text{end}(I)) = (M\ A[q]\ \text{start}(I))
\end{aligned}
$$

where

$$
\begin{aligned}
&\forall\, n \in \textit{numbers}\ \forall\, I \in \textit{time} \\
&\text{constant-sign}(n\ I) \equiv (\forall\, i_1, i_2 \in \text{during}(I)\ (M\ s[n]\ i_1) = (M\ s[n]\ i_2)).
\end{aligned}
$$

This statement is very weak compared to our usual notion of integrability.[11] In particular, it does not rest on knowing an explicit function describing the derivative and thus does not require an explicit notion of integral.

3.3. Limit points

Recall that a quantity space consists of a collection of elements and ordering relations between them. The major source of elements for the quantity space of some quantity Q are the numbers and constants that are compared to Q via quantity conditions. Because they correspond to discontinuous changes in the processes that are occurring (or individual views that hold), they are called *limit points*. Limit points serve as boundary conditions. For example, the temperature quantity space for an object W might include the limit points t-melt(W) and t-boil(W), where the object undergoes phase changes that result in qualitatively distinct behavior. These different modes of behavior are modeled by individual views.

3.4. The sole mechanism assumption and process vocabularies

The central assumption of qualitative process theory is the *sole mechanism* assumption, namely:

Sole mechanism assumption. *All changes in physical systems are caused directly or indirectly by processes.*

[11]If the time involved is an instant (i.e., an interval of duration ZERO), then we also assume that the quantity "doesn't change very much" during this time. To be more exact, we assume in that case the quantity is only different by an infinitesimal amount, or equivalently, that influences are finite. This assumption underlies case (2) of the equality change law, which is discussed shortly.

As a consequence, the physics for a domain must include a vocabulary of processes that occur in that domain. This *process vocabulary* can be viewed as specifying the dynamics theory for the domain. A physical situation, then, is described by a collection of objects, their properties, the relations between them (including individual views), *and* the processes that are occurring. The sole mechanism assumption allows us to reason by exclusion. If we make the additional assumption that our process vocabulary for a domain is complete, then we know what types of quantities can be directly influenced. If we understand the objects and relationships between them well enough, we know all the ways quantities can be indirectly influenced. Thus we know all the potential ways any physical situation will change. Without these closed-world assumptions (see [5, 35, 38]), it is hard to see how a reasoning entity could use, much less debug or extend, its physical knowledge.

3.5. Reprise

Processes should be first-class entities in the ontology of naive physics. It may be tempting to think that processes are mere abbreviations for 'deeper' representations, such as constraint laws. However, they are not. The temptation arises both because constraint laws are often judged to be the most elegant physical descriptions in 'non-naive' physics, and because constraint-based computer models have been fairly successful for analyzing engineered systems [9, 43]. However, the aims of naive physics are not the same as the aims of physics or engineering analysis. In physics we are trying to construct the simplest models that can make detailed predictions about physical phenomena. When performing an engineering analysis, even a qualitative one, we have chosen a particular point of view on the system and abstracted away certain objects. Unlike either of these enterprises, naive physics attempts to uncover the ideas of physical reality that people actually use in daily life. Thus the notions that physics throws away (objects, processes, causality) for conciseness in its formal theory—the equations—are precisely what we must keep.

Qualitative process theory concerns the form of dynamical theories, not their specific content. For example, the heat flow process illustrated previously adheres to energy conservation, and does not specify that 'stuff' is transferred between the source and destination. The language provided by the theory also allows one to write a heat flow process that violates energy conservation and transfers 'caloric fluid' between the source and destination. The assumptions made about the content of dynamical theories are quite weak. Aside from the ability to write a wide variety of physical models, the weakness of its assumptions allow other theories to be written that impose further constraints on the legal vocabularies of processes. For example, conservation of energy can be expressed as a theory about certain types of quantities and the allowable patterns of influences in processes that affect those types of quantities (see

Section 4.5). We do not, however, wish to saddle QP theory with these assumptions.

3.6. Basic deductions

To be useful, a representation must support deductions. Several basic deductions involving the constructs of QP theory are catalogued below. It may be helpful to skip momentarily to the example in Section 4.1, which illustrates these deductions step by step.

3.6.1. *Finding possible processes*

A process vocabulary specifies the types of processes that can occur. Given a collection of individuals and a process vocabulary, the individual specifications from the elements in the process vocabulary must be used to find collections of individuals that can participate in each kind of process. These process instances (PIs) represent the potential processes that can occur between a set of individuals. A similar deduction is used for finding view instances.

3.6.2. *Determining activity*

A process instance has a status of Active or Inactive according to whether or not the particular process it represents is acting between its individuals. By determining whether or not the preconditions and quantity conditions are true, a status can be assigned to each process instance for a situation.[12] The collection of active PIs is called the *process structure* of the situation. The process structure represents "what's happening" to the individuals in a particular situation. Similarly, the *view structure* is the collection of active VIs in the situation. Whenever we discuss the process structure, we usually include the view structure as well.

3.6.3. *Determining changes*

Most of the changes in an individual are represented by the D_s-values for its quantities. A D_s-value of -1 indicates the quantity is decreasing, a value of 1 indicates that it is increasing, and a value of 0 indicates that it remains constant. As stated previously, there are two ways for a quantity to change. A quantity can be directly influenced by a process, or it can be indirectly influenced via $\propto_Q$. (By the sole mechanism assumption, if a quantity is uninfluenced its D_s-value is 0.) Determining the D_s-value for a quantity is called *resolving* its influences, by analogy to resolving forces in classical mechanics.

Resolving a quantity which is directly influenced requires adding up the

[12]This can require searching the completions of the relevant quantity spaces if the required orderings cannot be deduced from what is already known about the value.

influences. If all the signs of the influences are the same then the D_s-value is simply that sign. Since we do not have numerical information, ambiguities can arise. Sometimes an answer can be found by sorting the influences into positive and negative sets and using inequality information to prove that one set of influences must, taken together, be larger than the other set. However, there is not always enough information to do this, so direct influences are not always resolvable.

Resolving an indirectly influenced quantity involves gathering the $\propto_Q$ statements that specify it as a function of other quantities. Because we lack detailed information about the form of the function, in many cases indirect influences cannot be resolved within basic QP theory. An example will make this point clearer. Suppose we have a quantity Q_0 such that in a particular process structure:

$$Q_0 \propto_{Q+} Q_1 \wedge Q_0 \propto_{Q-} Q_2 .$$

If we also know that

$$D_s[Q_1] = 1 \wedge D_s[Q_2] = 1$$

then we cannot determine $D_s[Q_0]$, because we do not have enough information to determine which indirect influence dominates. However, if we had

$$D_s[Q_1] = 1 \wedge D_s[Q_2] = 0$$

then we can conclude that

$$D_s[Q_0] = 1$$

because Q_1 is now the only active indirect influence.

Importantly, we assume the collection of qualitative proportionalities which hold at any particular time is loop-free, that is, if A is qualitatively proportional to B then it cannot also be the case that B is qualitatively proportional to A. At first glance it might seem that this assumption makes it impossible to model systems where two parameters are interdependent, such as feedback systems. This is not the case; the key observation is that, in physical systems, such loops always contain a derivative relationship—which is modeled by a direct influence, not a qualitative proportionality. In thinking about fluid flow, for example, we might observe that a change in amount of liquid causes a change in flow rate, which in turn acts to change the amount of liquid. But while flow rate is qualitatively proportional to the amount of liquid (via its dependence on pressure, which depends on the level, which in turn depends on the amount of liquid), the flow rate is a direct influence on the amount of liquid. The integral

connection between them serves to 'break' the loop, thus ensuring that the system of qualitative proportionalities is loop-free.

Domain-specific and problem-specific knowledge often plays a role in resolving influences. We may know that a certain influence can be ignored, such as when we ignore heat loss from a kettle on a stove to the air surrounding it if the stove is on. Our knowledge about particular functions may tell us which way things combine. Suppose for instance that our model of fluid flow included influences to model the changes in heat and temperature that result from mass transfer. In the source and destination temperature would be indirectly influenced (via Amount-of and heat), and if we knew only the D_s-values we could say nothing about how they will change. From Black's Law, however, we know that the temperature of the source is unchanged and the temperature of the destination will rise or fall according to whether the temperature of the source is greater or less than the temperature of the destination.

3.6.4. *Limit analysis*

Changes in quantities can result in the process and view structures themselves changing. Determining these changes and changes in D_s-values is called *limit analysis.* Limit analysis is carried out by using the current D_s-values and quantity spaces to determine how the quantity spaces (and hence the truth of quantity conditions) can change.

The first step is to find the neighboring points within the quantity spaces of each changing quantity. If there is no neighbor in some direction, then a change in that direction cannot affect the status of any process. The ordering between each neighbor and the current amount of the quantity can be combined with the D_s-values of each to determine if the relationship will change (see Fig. 12). If the neighbor is a limit point, some processes may end there and others begin. Thus the set of possible changes in orderings involving limit points determines the ways the current set of active processes might change.[13] The set of changes between single inequalities plus consistent conjunctions of changes (corresponding to the occurrence of simultaneous changes) forms the set of *quantity hypotheses* for the current situation. A quantity hypothesis that imposes a change in either the view or process structure (as opposed to merely indicating a change in a D_s-value) will be called a *limit hypothesis.*

Determining which changes and conjunctions of changes are consistent involves several types of knowledge. First, one change might render another

[13]This assumes that rates are not infinitesimals, so that if a quantity is 'moving' towards some point in its quantity space it will actually reach that value in some finite time. This assumption rules out a simple form of Zeno's paradox. Note, however, that relaxing this assumption would result in only one additional state in the possibilities returned by the limit analysis—that the current set of active processes never changes.

For A > B:

A \ B	−1	0	1
−1	N1	=	=
0	>	>	=
1	>	>	N2

N1: If $D_m[A] > D_m[B]$ then =; > otherwise
N2: If $D_m[A] < D_m[B]$ then =; > otherwise

For A = B:

A \ B	−1	0	1
−1	N3	<	<
0	>	=	<
1	>	>	N4

N3: If $D_m[A] > D_m[B]$ then <;
If $D_m[A] < D_m[B]$ then >;
If $D_m[A] = D_m[B]$ then =;

N4: If $D_m[A] > D_m[B]$ then >;
If $D_m[A] < D_m[B]$ then <;
If $D_m[A] = D_m[B]$ then =;

FIG. 12. Linking derivatives with inequalities. This table summarizes how the ordering relationship between two quantities may change according to the sign of their derivatives over some interval.

change moot. For example, if a particular change causes an individual to vanish, then any other changes involving that individual are irrelevant. Second, we assume that changes must be continuous both in quantity spaces and in D_s-values. Continuous in quantity spaces means that all order relations must go through equality, i.e., that the relationship between N_1 and N_2 cannot change directly from > to < or from < to >. Continuous in D_s-values means a D_s-value cannot jump directly from 1 to −1 or from −1 to 1. Finally, domain-dependent information can be used to determine that the situation resulting from the change is inconsistent. For example, if the bottoms of two open containers are at the same height and the only thing happening is a fluid flow from one to the other, then it is impossible for the source of the flow to run out of liquid.

Typically more than one change is possible, as the examples in the next section illustrate. There are three reasons for this. First, if the ordering within a quantity space is not a total order more than one neighbor can exist. Second, a process can influence more than one quantity. Finally more than one process can be occurring simultaneously. The basic theory does not in general allow the determination of which alternative actually occurs. Using calculus as the model for quantities, the alternative which occurs next is the one for which time to

integrate the quantities involved to their limit points is minimal. Since the basic theory does not include explicit integrals, this question typically cannot be decided.

There are some special situations, due to the nature of quantities, where sometimes we can rule out classes of hypotheses without detailed domain-specific information. Consider two quantities A and B that are equal, and C and D that are unequal. If all of the quantities are changing (D_s-value of -1 or 1) in ways that insure the relationships between them will change, then the finite difference between C and D implies that the change in the equality between A and B occurs first. In fact, we assume that the change *from* equality occurs in an instant, while the change *to* equality usually will take some interval. We further assume that the only time a change *to* equality will take an instant is when the change in value was due to a process that acted only for an instant. We will summarize this as the *equality change law*:

Equality change law. *With two exceptions, a process structure lasts over an interval of time. It lasts for an instant only when either*

(1) *a change from equality occurs, or*

(2) *a change to equality occurs between quantities that were influenced away from equality for only an instant.*

The first case assumes that the values of numbers are not 'fuzzy', and the second case assumes that the changes wrought by processes are finite (i.e., no impulses).

Remember that the set of quantity hypotheses consists of single changes and conjunctions of single changes. Consider the set of conjunctive hypotheses which contain only changes that occur in an instant, and in particular, the maximal element (in terms of inclusion) of the set. The quantity hypotheses that contain this maximal element are the ones which can occur next, because the duration of an instant is shorter than the duration of an interval. By using the equality change law to identify those quantity hypotheses that represent changes that occur in an instant, we can sometimes get a unique result from limit analysis within the basic theory.

For some kinds of tasks just knowing the possible changes is enough (e.g., envisioning). If required, knowledge outside the scope of QP theory can be used to disambiguate the possibilities. Depending on the domain and the style of reasoning to be performed there are several choices; among them simulation [2], algebraic manipulation [6], teleology [7], or possibly default assumptions or observations [17].

3.7. Processes and histories

Adding processes to the ontology of naive physics requires extending the history representation of change. In addition to parameter histories, we will

also use *process histories* to describe what processes are occurring when. The temporal extent of a process episode is the maximal time during which the status of the instance is constant, and the spatial extent is the spatial extent of the individuals involved in it. The events that bound episodes in the process history occur at the instants at which quantity conditions, preconditions, or the existence of objects involved in the instance change. *View histories*, describing the status of view instances, are defined similarly. Process and view episodes are included in the histories of the objects that participate in the process, and the union of the object's parameter histories and the histories of the processes and views it participates in comprise its total history. Fig. 13 illustrates the full

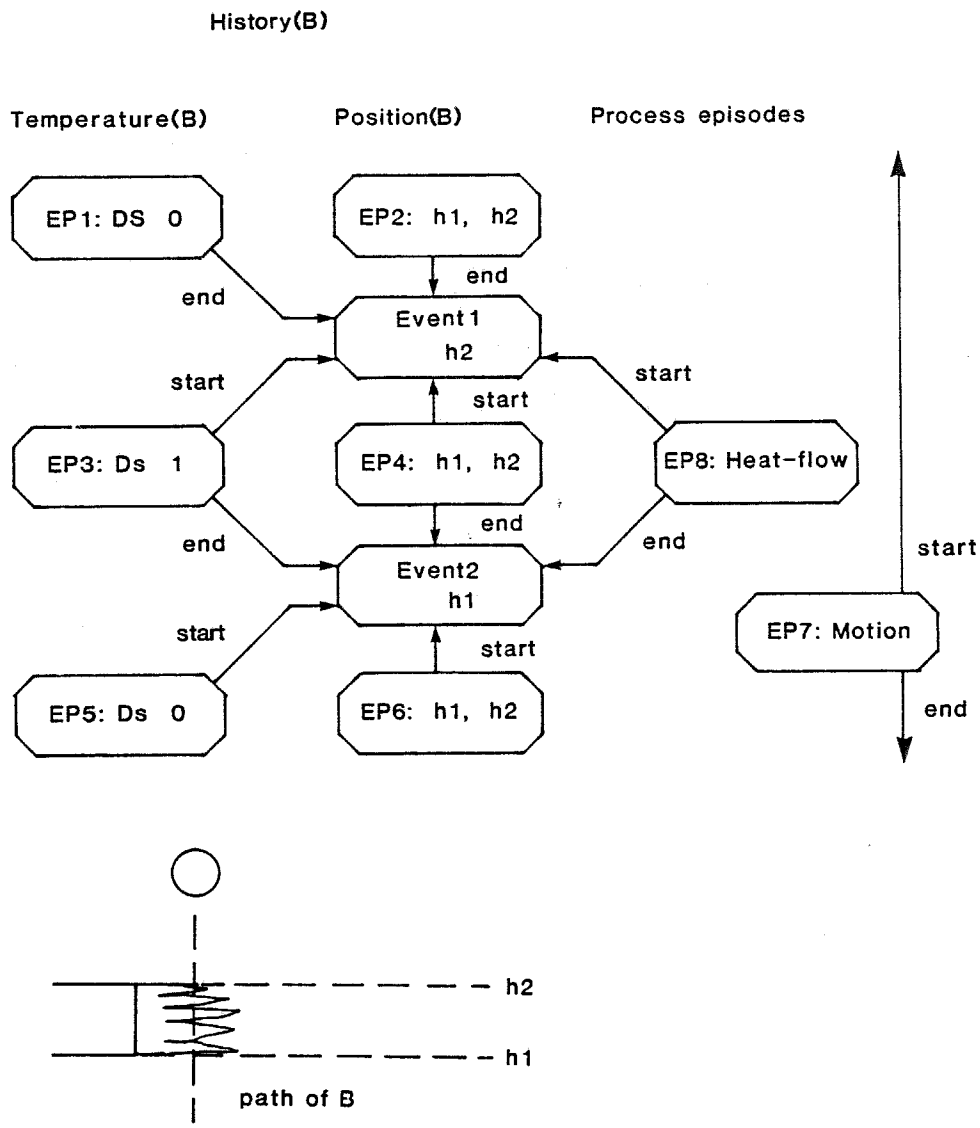


FIG. 13. History for a ball dropping through a flame. A piece of the history for the ball again, but with process episodes added. As before, EP⟨n⟩ are episodes, and time runs from top to bottom.

history over a small interval for the ball being dropped through a flame discussed previously.

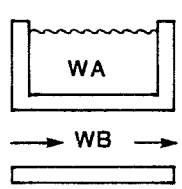
As mentioned previously, the two key problems in reasoning with histories are the *local evolution* problem (extending the known portion of an object's history, preferably by carving up the situation into pieces that can be reasoned about semi-independently) and the *intersection/interaction* problem. The key to solving them lies in having explicit descriptions of the ways changes are caused.

Recall that the processes active in a situation form its process structure (as usual, we also implicitly include the view structure to simplify discussion). Processes interact by shared influences; two processes which affect the same parameter or a process that affects a parameter mentioned in the quantity conditions of another must be considered together when figuring out if, and how, they will change. If there is no way for two processes to 'communicate' by common effects, then they can be considered independently. This suggests carving up what is happening at a particular time into 'non-overlapping' pieces, subsets of the process structure that do not interact.

We define *p-components* as equivalence classes on the process structure as follows. A process instance P1 is in the same p-component as another process instance P2 (or view instance) if either: (a) P1 influences a quantity mentioned in P2's quantity conditions, (b) P1 influences a quantity influenced by P2, (c) P1's quantity conditions mention a quantity mentioned in the quantity conditions of P2, or (d) P2 contains a $\propto_Q$ that propagates an influence of P1.

As long as a particular process structure lasts, the p-components can be reasoned about independently. For example, we usually don't worry about getting our feet wet in a basement despite the proximity of flowing water and steam in our plumbing. Changes in the process structure can bring about changes in p-components, so the conclusions made in each p-component may have to be modified depending on how the process structure changes. If our plumbing leaks, for instance, there are now ways for our feet to get wet.

The individuals affected by the processes in each p-component define a



If shared wall is not a heat path,
  PS: Fluid-Flow(WB, channel), no interaction
Otherwise, if A[Temperature(WA)] = A[Temperature(WB)]
  PS: Fluid-Flow(WB, channel), no interaction
  Otherwise, PS: Fluid-Flow(WB, channel) and a heat flow, hence they interact.

FIG. 14. Determining interactions. Suppose WA and WB are liquids, with WB the fluid flowing through the channel below WA's container. The process structures that result from different assumptions about the situation are shown, with potential interactions indicated.

collection of objects that can be reasoned about in isolation, barring certain changes in process structure. Thus we can generate object histories by evolving situations that correspond to p-components, combining the results when the process structure changes to get new p-components, and so forth. The interaction part of the intersection/interaction problem becomes trivial—two episodes interact if and only if the processes that give rise to them are part of the same p-conponent of a process structure on a situation made up of slices from those particular episodes. Fig. 14 provides a graphical illustration.

3.8. A language for behavior

QP theory concerns the structure of qualitative dynamics. We can view it as specifying a language in which certain commonsense physical models can be written. Can this language be extended to form a full language of behavior for physical systems? Although I have not yet done so, I will argue that the answer is yes, and that several advantages would result from the extension.

A language should have primitives, a means of combining these primitives, and some means of abstraction to allow new entities to be defined. Processes and individual views are obviously the primitives in this language.[14] There are two sensible kinds of compound processes. The first kind consists of processes that form a p-component, a shared-parameter combination. An example of a shared-parameter combination is the intake stroke of a four-cycle engine, which consists of a flow of air and gas into a cylinder and motion of the piston. The second kind consists of sequences of processes occurring over the same individuals. An example of a sequential combination is the sequence of intake, compression, combustion and exhaust strokes of a four-cycle engine. Treating these combinations as new 'things' then allows properties of the system they describe to be reasoned about.

It should be clear that the shared-parameter combination can be treated exactly as a simple process, specified by the union of the properties of the component processes. In other words, a shared-parameter combination will have individuals, preconditions, quantity conditions, relations, and influences that work just like any other process. However, the sequential combination is *not* a process, because the same influences and relations do not hold over every distinct time within the occurrences of the sequential combination. A sequential combination is really a piece of a history! In particular, it is an abstraction of the history of the individuals affected by the processes, viewed as a system.

[14]The choice of what is primitive in any particular domain's vocabulary will of course vary—for example, the description of a gas we use later is macroscopic. Presumably a richer process vocabulary would contain the 'mechanisms' that induce these relations (i.e., the kinetic theory of gases), but there is no reason to always include such detail. Consider for example a resistor in a circuit that never exceeds its electrical capacity. The detailed mechanics of conduction hinder rather than help when calculating the current that will result from a voltage across it.

In honor of this mixed ontological status such descriptions will be called *encapsulated histories.* Encapsulated histories (EH) are important for two reasons. First, some phenomena which can be described in that form seem irreducible in terms of processes—collisions, for example. Second, they serve as abstract descriptions for more complex behavior, e.g. in describing the pattern of activity in an oscillator.

When writing encapsulated histories, we will use most of the syntactic structure of processes and individual views, in that the combination will have individuals, preconditions, quantity conditions, and relations. However, the relations component is restricted to holding a description of a piece of the history for the individuals, and the preconditions and quantity conditions are written relative to episodes in that piece of history. If the preconditions and quantity conditions are ever true for a partial history of a collection of objects matching the individual specifications, then the schematic history described in its relations is instantiated as part of the history of those objects.[15] We will see colliding described as an encapsulated history in Section 4.3.

For those phenomena which are irreducible, the encapsulated history may be the only way to evolve the history of the object past that point. For systems where the encapsulated history serves as a summary, an interesting kind of perturbation analysis becomes possible (as we will see in Section 4.5.1). In performing an energy analysis, for example, the quantity conditions are rewritten in terms of energy. Changes to the system, such as adding friction, are modeled by processes that influence energy, and the effects of these changes are determined by examining the episodes that comprise the encapsulated history.

3.9. Classification and abstraction

A classification hierarchy is needed to account for the various kinds of conditions under which processes occur. For example, Hayes [22] elucidates several distinct conditions under which fluid flow occurs. Another example is the process of motion—flying, sliding, swinging, and rolling are distinct types of motion, despite sharing certain common features. Sliding and rolling are examples of motion along a surface, and along with swinging comprise the motions involving constant contact with another object. Each of these conditions has slightly different properties, but they are sufficiently similar in the individuals they involve and the pattern of influences they engender to be considered the same kind of process. Having explicit abstract descriptions of processes should also be useful because they are often easier to rule out than more detailed descriptions. If, for instance, there is no path between two places

[15]Many of diSessa's "phenomenological primitives" [10] appear to be representable as encapsulated histories. Encapsulated histories are also good candidates for the first models people make of a new domain [19].

through which an object can be moved, it cannot get there by sliding, flying, rolling, or any other kind of motion that might exist.

Theoretically, disjunctions could be used within a single-process description to cover the various cases. Doing so would lead to complicated descriptions that could not easily be reasoned about. Instead, every case will be represented by a different process. We will indicate that P1 is a case of P2, such as:

Case-of(Swinging, Motion) .

The following restrictions hold on cases:

Specificity: There is a subset of the individuals specified for P1 such that they or individuals whose existence is implied by them match the individual specifications of P2. The preconditions and quantity conditions for P1 imply the preconditions and quantity conditions for P2 respectively.

Inheritance: All statements in the relations and influences fields of P2 hold for P1 unless explicitly excluded.

Fig. 15 illustrates some specializations of the abstract motion process that will be discussed in Section 4.3.

```
Process Motion(B, dir)

Individuals:
  B an Object, Mobile(B)
  dir a direction

Preconditions:
  Free-direction(B, dir)
  Direction-Of(dir, velocity(B))

QuantityConditions:
  A_m[velocity(B)] > ZERO

Influences:
  I+ (position(B), A[velocity (B)])
```

```
Process Slide                 Process Roll

Case-of: Motion               Case-of: Motion

Individuals:                  Individuals:
  S a surface                   S a Surface

Preconditions:                Preconditions:
  Sliding-Contact(B, S)         Contact(B, S)
  AlongSurface(dir, B, S)       Round(B)
                                AlongSurface(dir, B, S)
```

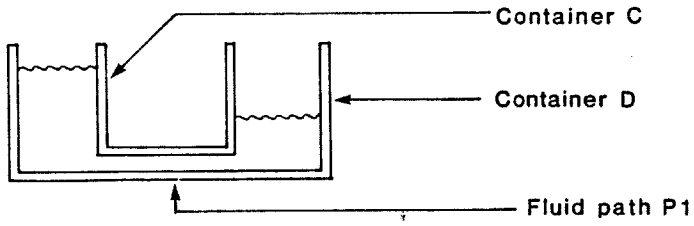
FIG. 15. Some specialized descriptions of motion. Cases of motion are organized around constraints on kinematics. The abstract motion process already includes the individuals B, a movable object, and dir, a direction. The abstract motion process will be explained in more detail later. In sliding and rolling there is contact with a surface, but different constraints on the kind of contact. Otherwise the same facts pertain to them as to the abstract version of motion.

4. Examples

At this point a great deal of representational machinery has been introduced. It is time to illustrate how QP theory can be used in physical reasoning. The examples are fairly informal for two reasons. First, the formalization of the domains is still underway.[16] Second, while QP theory provides the means to represent an important part of a domain's theory, a complete model usually has to address several considerations besides dynamics, such as spatial reasoning (qualitative kinematics, as it were). Still, these examples are complex enough to provide an indication of the theory's utility. The assumptions about other kinds of required knowledge are noted as they occur.

4.1. Modeling fluids and fluid flow

This example illustrates some of the basic deductions sanctioned by qualitative process theory and introduces the representations of fluids that are used in other examples. These representations are slightly more complex than the contained-liquid description we have been using. Consider the two containers illustrated in Fig. 16. What will happen here?




```
; structural description

Open-Container(C)
Open-Container(D)
Fluid-Path(P1)
Fluid-Connected(C, D, P1)

; some substances are in the containers
Contains-Substance(C, water)
Contains-Substance(D, water)

; the levels are related
(M Level-in(C, water) Initial) > (M Level-in(D, water) Initial)
```

FIG. 16. Two partially filled containers. Containers C and D are connected by a pipe. Initially C contains more water than D. In general, an "-in" suffix indicates a function that maps from a container and a substance to a quantity.

[16]At present work is focusing on two domains: the *mechanism world*, and the *fluids world*. The mechanism world includes the blocks world but also more complex shapes and some non-rigid materials. The aim of work in the mechanism world is to understand devices such as mechanical watches and automobile transmissions. The fluids world is an attempt to extend Hayes' theory of liquids to include gases and more complex fluid systems such as found in steam plants.

We first introduce descriptions of the fluids. Following Hayes [22], we individuate liquids according to what contains them. Fig. 17 describes 'pieces of stuff', independently of the particular criteria used for individuating them. Fig. 18 describes a particular class of pieces of stuff that are individuated by being the stuff inside a container. Any piece of stuff must be in some state, either solid, liquid, or gas. Fig. 19 describes the states of substances. The interaction of state and containment is described in Fig. 20. Since the containers initially contain some water, we will create individuals corresponding to the water in

```
; A piece of stuff consists of several quantities, a substance, and
; a location
(∀ p ∈ piece-of-stuff
  Has-Quantity(p, amount-of)
  ∧ Has-Quantity(p, volume) ∧ Has-Quantity(p, pressure)
  ∧ Has-Quantity(p, temperature) ∧ Has-Quantity(p, heat)
  ∧ Substance(made-of(p)) ∧ Place(location(p))
  ∧ temperature(p) ∝Q+ heat(p))

; where
(∀ p ∈ things ∀ q ∈ quantity-type
  Has-Quantity(p, q) ↔ Quantity(q(p)))
```

FIG. 17. Pieces of stuff. A piece of stuff is mainly described by several quantities and its location.

```
Individual-View Contained-Stuff

Individuals:
  c a container
  s a substance

Preconditions:
  Contains-Substance(c, s)

QuantityConditions:
  A[amount-of-in(c, s)] > ZERO

Relations:
  There is p ∈ piece-of-stuff
  amount-of-in(c, s) = amount-of(p)
  s = made-of(p)
  inside(c) = location(p)
```

FIG. 18. Contained stuff. Contained-Stuff describes the conditions under which pieces of stuff exist inside a container.

each container. Call the pieces of stuff in containers C and D WC and WD respectively. We will assume their temperatures are such that they are both liquids. For simplicity we will ignore the liquid in the pipe P1. We will also ignore the precise definition of fluid paths, except to note that P1 is one, connecting the two contained fluids.

```
Individual-View Solid(p)

Individuals:
  p a piece-of-stuff

QuantityConditions:
  ¬ A[temperature(p)] > A[t-melt(p)]
  ¬ Liquid(p)
```

```
Individual-View Liquid(p)

Individuals:
  p a piece-of-stuff

QuantityConditions:
  ¬ A[temperature(p)] < A[t-melt(p)]
  ¬ A[temperature(p)] > A[t-boil(p)]
  ¬ Solid(p)
  ¬ Gas(p)

Relations:
  volume(p) ∝Q+ amount-of(p)
  t-boil(p) ∝Q+ pressure(p)
```

```
Individual-View Gas(p)

Individuals:
  p a piece-of-stuff

QuantityConditions:
  ¬ A[temperature(p)] < A[t-boil(p)]
  ¬ Liquid(p)

Relations:
  temperature(p) ∝Q+ pressure(p)
  pressure(p) ∝Q+ amount-of(p)
  pressure(p) ∝Q- volume(p)
  pressure(p) ∝Q+ heat(p)
```

FIG. 19. States of matter. The temperatures at which state changes occur are modeled by two functions t-melt and t-boil. t-melt and t-boil map pieces of stuff onto quantities, and we assume A[t-boil] is never less than A[t-melt]. The quantity conditions express the fact that a substance can be in either state at a phase boundary, but that a particular piece cannot be in both states at once. To determine the state of a piece of stuff at the phase boundary requires either knowing its history or making an assumption.

```
; Contained stuff has states as well –

(∀ p ∈ piece-of-stuff
   (Contained-Gas(p) ↔ (Contained-Stuff(p) ∧ Gas(p)))
   ∧ (Contained-Liquid(p) ↔ (Contained-Stuff(p) ∧ Liquid(p)))
   ∧ (Contained-Solid(p) ↔ (Contained-Stuff(p) ∧ Solid(p))))

; Contained liquids have levels, which are tied to amounts
; and in turn (assuming an open container) determines pressure

(∀ c ∈ contained-liquid
   Has-Quantity(c, level)
   ∧ level(c) ∝Q+ amount-of(c)
   ∧ (Open-Container(space-of(location(c))) ⇒
        Function-Spec(p-l-fun, {pressure(c) ∝Q+ level(c)})))
```

FIG. 20. Effects of state on containment.

Suppose our process vocabulary consists of fluid flow, whose description is illustrated in Fig. 21. This model is very simple, because it ignores the possibility of different kinds of fluids and the details of how fluids move through the fluid paths ([22] illustrates some of the distinctions that should be drawn in a more detailed model).

With the situation we have so far, there are two process instances, one corresponding to flow from WC to WD and the other corresponding to flow from WD to WC. To determine if either is active (thus determining the process structure) we have to know the relative pressures of WC and WD. Assume we deduce from the relative levels that the pressure of WC is greater than the pressure of WD. Then the process instance representing fluid flow from WC to WD will be active, and the process instance representing fluid flow from WD to WC will be inactive. Thus the process structure is the set consisting of Fluid-Flow(WC, WD, P1).

To find out what changes are occurring we must resolve the influences. In this situation resolving influences is simple. The fluid flow from C to D is the only cause of direct influences, changing amount-of for WC and WD. Each of

```
process fluid-flow

Individuals:
  src a contained-liquid
  dst a contained-liquid
  path a fluid path, Fluid-Connected(src, dst, path)

Preconditions:
  Aligned(path)

QuantityConditions:
  A[pressure(scr)] > A[pressure(dst)]

Relation:
  Let flow-rate be a quantity
  flow-rate ×Q+ (A[pressure(src)] − A[pressure(dst)])

Influences:
  I+ (amount-of(dst), A[flow-rate])
  I− (amount-of(src), A[flow-rate])

; A fluid path is aligned if only if either it has no valves or every valve is open
(∀ p ∈ fluid-path
  ((number-of-valves(p) = 0) ⇒ Aligned(p))
  ∧ ((number-of-valves(p) > 0) ⇒ (∀ v ∈ valves(p) Open(v)) ↔ Aligned(p))
  ∧ ¬ (number-of-valves(p) < 0))

; A heat path is defined in terms of objects in contact, and aligned
; indicates that the contact is unbroken. We ignore heating
; by radiation and by convection, as well as temperature changes
; caused by mixing.
```

FIG. 21. A process description of fluid flow. This simple model does not describe the existence and behavior of the fluid within the fluid path.

them has only one influence, hence

$$D_s[\text{amount-of(WC)}] = -1 \quad \text{and} \quad D_s[\text{amount-of(WD)}] = 1\,.$$

These in turn influence volume, level and pressure, each of which has only one $\propto_Q$ applicable (see Fig. 20). Thus we can deduce that the volume, level and pressure of WC are all decreasing, and the volume, level and pressure of WD are all increasing. All other quantities are uninfluenced, hence unchanging. Limit analysis is similarly simple. The pressures will eventually be equal, which means the fluid flow will stop. It is also possible that the container C will run out of water, thus ending WC's existence (although it is not possible in the particular drawing shown). These results are summarized in Fig. 22. This graph of process structures can be used to generate a history by first creating the appropriate episodes for objects and processes from their initial slices, and then selecting one or the other limit hypothesis as the end-event for that episode. Usually we will just represent the interconnections between possible process structures as we have done here. With only a single process and simple relationships between quantities, resolving influences and limit analysis are simple. In more complex situations resolving influences and disambiguating the possibilities raised by limit analysis will require more information, as we will see below.

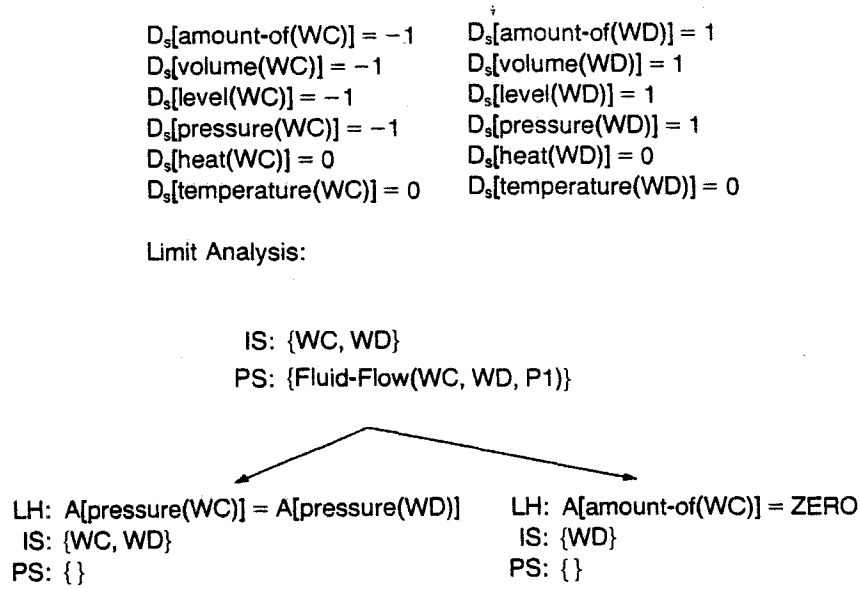


FIG. 22. Resolved influences and limit analysis. The results of resolving influences and limit analysis for the situation involving two containers are summarized. The individuals in the situation are labeled IS, the process structure by PS, and limit hypotheses by LH.

4.2. Modeling a boiler

Let us consider the possible consequences of the situation shown in Fig. 23. The situation consists of a container partially filled with water. Initially the lid of the container is open; we stipulate that if boiling ever occurs, the lid will be closed and sealed. A flame, modeled as a temperature source, is placed so that heat can be conducted to the container and water (i.e., there is an aligned heat path between them). What sorts of things can happen?

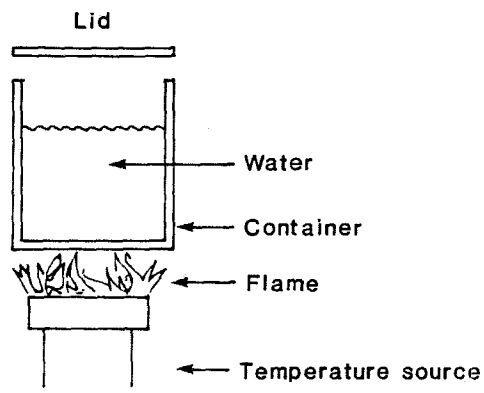


FIG. 23. A simple boiler.

To begin with, we need the contained substances defined in the previous example and a model of containers. We assume that if the pressure inside the container exceeds a particular pressure p-burst(CAN), the container will explode. Fig. 24 describes the container model. We will assume that, in addition to fluid flow, the process vocabulary includes heat flow and boiling, as presented in Section 3.1. We will ignore the rest of the details, such as the nature of heat and fluid paths and the detailed geometry of containers.

We start by assuming that no processes are active before the heat source is turned on; in other words that all temperatures, pressures, etc. are equal so there are no flows, and that the temperatures are in the appropriate regions of their quantity spaces so that no state changes are occurring. (Note that, as usual, we are making a closed-world assumption both in assuming our process vocabulary is complete and that we know all of the relevant individuals.) Since there is a heat path between the source and the container, if we turn the heat source on and if

$$A[\text{temperature(SOURCE)}] > A[\text{temperature(WATER)}]$$

there is a heat flow from the source to the water. We ignore the influence of the heat flow on the source by assuming

$$D_s[\text{temperature(SOURCE)}] = 0\,.$$

```
∀ c ∈ container
  [Has-Quantity(c, volume) ∧ Has-Quantity(c, pressure)
   ∧ Has-Quantity(c, temperature) ∧ Has-Quantity(c, heat)
   ∧ (Rigid(c) ⇒ Ds[volume(c)] = 0)
   ∧¬Open-Container(c) ⇒
      (∀ p ∈ contents(c)
         pressure(c) = pressure(p)
          ∧ temperature(c) = temperature(p))]

; note we are assuming instantaneous equilibration
; within the container

Encapsulated History Explode

Individuals:
  c a container, rigid(c)
  e an episode

Preconditions:
  (T ¬ Open-Container(c) e)

QuantityConditions:
  (M A[pressure(c)] end(e)) = (M A[p-burst(c)] end(e))
  (M A[pressure(c)] during(e)) < (M A[p-burst(c)] during(e))

Relations:
  Let EV1 be an event
  end(e) = EV1
  (T Explodes(c) EV1)
  Terminates(c, EV1)

; The relation Terminates indicates that the object does not exist past
; that particular event
```

FIG. 24. A simple container model. For simplicity we will model a container only as a collection of quantities, a set of pieces of stuff which are its contents, and an encapsulated history to describe the possibility of it exploding. The geometric information necessary to determine flow paths and the spatial arrangement of the contents will be ignored.

The only influence on temperature(CAN) is that of the heat flow, so

$$D_s[\text{temperature(CAN)}] = 1 .$$

This in turn causes a heat flow to the air surrounding the container and to the air and the water inside the container. Since we are only thinking about the container and its contents the loss of heat to the air will be ignored, and from now on when we refer to heat flow it will be the flow from the flame to the contained stuff, using the container as the heat path. The temperature quantity space that results is illustrated in Fig. 25. If A[temperature(SOURCE)] > A[t-boil(WATER)] and the process is unimpeded (i.e., the preconditions for the heat flow remain true), the next process structure to occur will include a boiling.

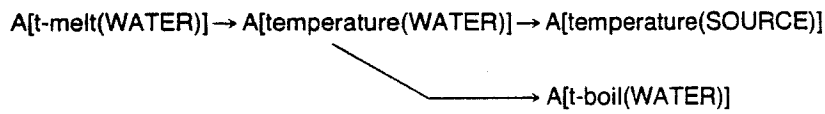


FIG. 25. Quantity space for water temperature. The heat flow is increasing the heat, and thus (via $\propto_{Q+}$) the temperature of the water. The lack of ordering information between the temperature of the source and the boiling temperature leads to uncertainty about what will occur next.

Suppose the preconditions for the heat flow continue to be met and boiling occurs. Then by our initial assumptions the lid will be sealed, closing all fluid flow paths and thus preventing any flows. The amount-of quantity spaces that result are illustrated in Fig. 26. The influence of the boiling on amount-of(WATER) moves it towards ZERO. So one of the ways the process structure might change is that all of the water is converted to steam.

If all the water is converted to steam, the only active process is a heat flow from the heat source to the steam. Thus the sole influence on the heat of the steam is positive, and (because of $\propto_Q$) the temperature also rises. Heat indirectly influences pressure, so the pressure of the steam will also rise. By examining the quantity spaces for temperatures and pressures we find there are two limit points which may be reached, namely that the temperature of the steam can reach the temperature of the heat source and that the pressure of the container (which is equal to the pressure of the steam) can reach the explosion point. In the first case there are no active processes, and in the second an explosion occurs. We have found one possible disaster, are there more? To find out, we must go back to the boiling episode and check the indirect consequences of the changes in amount-of(STEAM).

ZERO → A[amount-of(WATER)]

ZERO → A[amount-of(STEAM)]

FIG. 26. amount-of quantity spaces.

Consider some arbitrary instant I within the boiling episode. Because the steam is still in contact with the water their temperatures will be the same. Since we assumed the container would be sealed when boiling began, there are no fluid paths hence no fluid flows. Therefore during I the only influence on amount-of(STEAM) and on amount-of(WATER) is from boiling. So

$$D_s[\text{amount-of(STEAM)}] = 1 \quad \text{and} \quad D_s[\text{amount-of(WATER)}] = -1\,.$$

Because steam is a gas, there are several indirect influences on temperature(STEAM) and pressure(STEAM) (see Fig. 19). In particular,

$$\text{temperature(STEAM)} \propto_{Q+} \text{pressure(STEAM)}$$
$$\text{temperature(STEAM)} \propto_{Q+} \text{heat(STEAM)}$$
$$\text{pressure(STEAM)} \propto_{Q+} \text{amount-of(STEAM)}$$
$$\text{pressure(STEAM)} \propto_{Q-} \text{volume(STEAM)}$$
$$\text{pressure(STEAM)} \propto_{Q+} \text{heat(STEAM)}$$

Assuming the container is rigid, $D_s[\text{volume(CAN)}] = 0$, and since the spaces of the steam and and water are separate and fill the container,

$$\text{volume(CAN)} = \text{volume(WATER)} + \text{volume(STEAM)} .$$

Since

$$D_s[\text{volume(WATER)}] = -1 , \qquad D_s[\text{volume(STEAM)}] = 1$$

and

$$D_m[\text{volume(STEAM)}] = D_m[\text{volume(WATER)}] .$$

Assume the function that determines pressure(STEAM) is continuous in amount-of(STEAM), heat(STEAM), and volume(STEAM). Then for any particular D[amount-of(STEAM)] and D[heat(STEAM)], we can find a corresponding D[volume(STEAM)] such that

$$(\text{M } D_s[\text{pressure(STEAM)}]\ \text{I}) = 0$$

i.e., the pressure at the end of I will be the same as it was at the start of I. Let β stand for that value of D[volume(STEAM)]. Then

$$(\text{M A[volume(STEAM)] end(I)}) = (\text{M A[volume(STEAM)] start(I)}) + \beta$$

is necessary for D_s[pressure(STEAM)] to be zero. A fact about steam is that, at any particular pressure and temperature, the volume of steam is very much greater than the volume of water it was produced from.[17] In other words,

$$D_s[\text{pressure(STEAM)}] = 0 \Rightarrow D_m[\text{volume(WATER)}] \ll D_m[\text{volume(STEAM)}] .$$

But in fact,

$$D_m[\text{volume(STEAM)}] = D_m[\text{volume(WATER)}] ,$$

[17]At standard temperature and pressure, about 220 times greater in fact.

so

$$D[\text{volume(STEAM)}] < \beta.$$

This means that (M A[volume(STEAM)] end(I)) will be less than

$$(M\ A[\text{volume(STEAM)}]\ \text{start(I)}) + \beta\ ,$$

and because

$$\text{pressure(STEAM)} \propto_{Q-} \text{volume(STEAM)}\ .$$

the pressure of the steam will be greater than it was, i.e.,

$$D_s[\text{pressure(STEAM)}] = 1\ .$$

Since

$$D_s[\text{heat(STEAM)}] = 1\ ,$$

both of the influences on temperature(STEAM) are positive, so

$$D_s[\text{temperature(STEAM)}] = 1\ .$$

So far we have discovered that

$$D_s[\text{pressure(STEAM)}] = D_s[\text{temperature(STEAM)}] = 1\ .$$

Since the water and steam are in contact their pressures will be equal, and since pressure indirectly affects the boiling temperature, the boiling temperature will also rise. The possible relative rates introduce three cases. If the boiling temperature is rising faster, i.e.,

$$(D_m[\text{t-boil(WATER)}] > D_m[\text{temperature(STEAM)}])$$

then the boiling will stop, the heat flow will increase heat(WATER) again, the temperature will rise, and the boiling will begin again.[18] In the other two cases

$$(D_m[\text{t-boil(WATER)}] = D_m[\text{temperature(STEAM)}]$$

[18]The astute reader will notice that this situation gives rise to a cycle of states that corresponds to a rising equilibrium rather than an oscillation. We will discuss how to use the equality change law to distinguish between these cases in Section 5.1.

and

$$D_m[\text{t-boil(WATER)}] < D_m[\text{temperature(STEAM)}])$$

the boiling will continue, albeit at a higher temperature and pressure. In all three cases, the increasing pressure makes

$$A[\text{pressure(CAN)}] = A[\text{p-burst(CAN)}]$$

possible, in which case the container explodes. The alternatives are summarized in Fig. 27. To actually determine which of these occurs requires more information, but at least we have a warning of potential disaster.

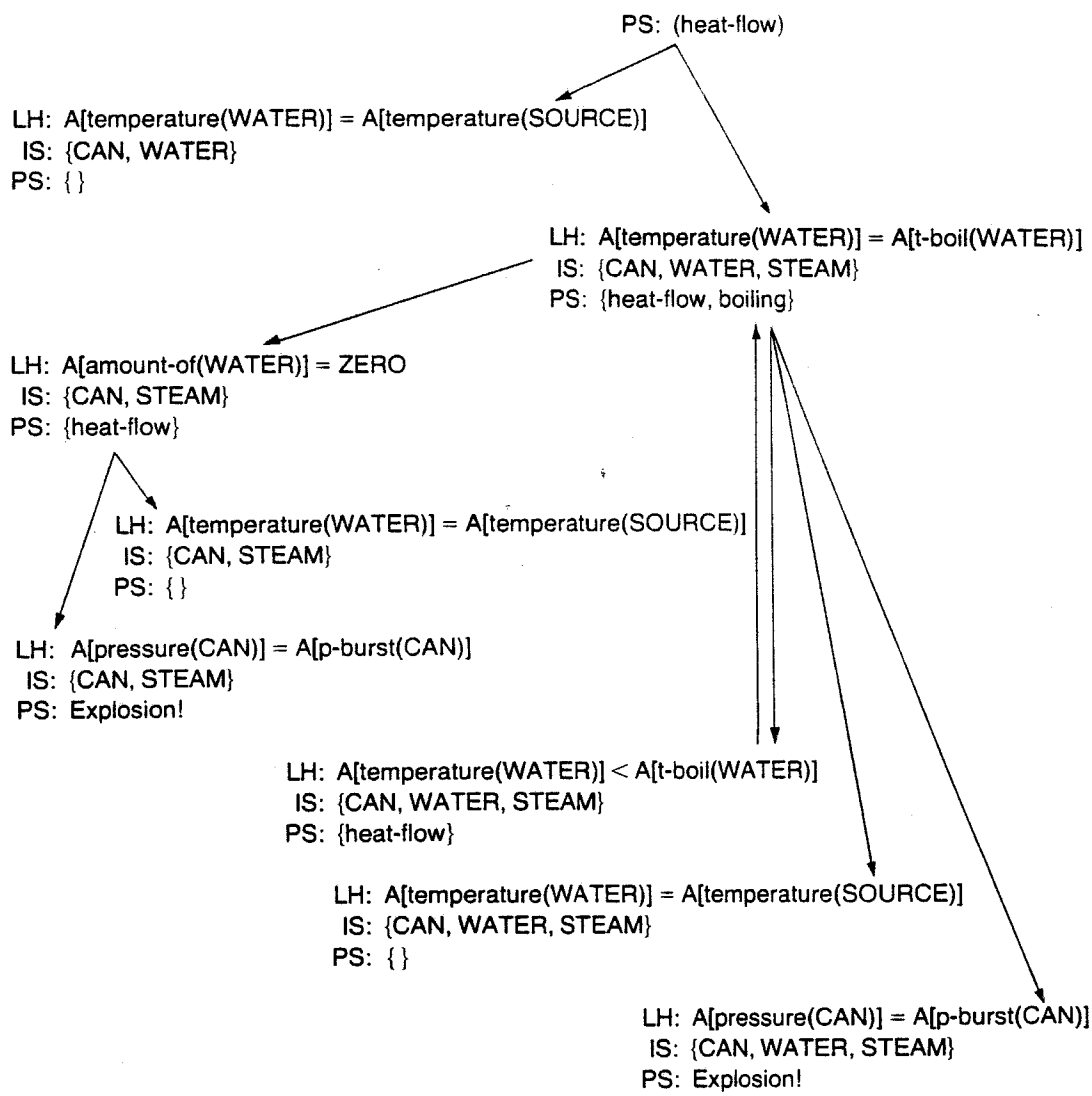


FIG. 27. Alternative behaviors for the boiler. The process structures are envisioned for water being heated in a sealed container, generated by repeated limit analysis.

4.3. Modeling motion

One process we reason about every day is motion. Motion is complex because it is intimately connected with our concepts of space and shape. Since QP theory describes the form of qualitative dynamics theories, it can only carry part of the representational burden imposed by motion. After developing a simple motion vocabulary, we compare the QP descriptions with the earlier qualitative-state representation in order to illustrate the strengths and weaknesses of the QP model.

4.3.1. *A simple motion vocubulary*

Consider a single object moving in one dimension. By ignoring the particular kind of motion it exhibits (FLY, SLIDE, SWING, ROLL) which depends on the particular shape and type of contact with other surfaces, we can develop an abstract vocubulary for describing motion. While very weak, such abstract descriptions have certain uses—we can deduce that if we kick something but it is blocked, for instance, then it will not move, and by ruling out most abstract motion possible, we rule out all the more specific kinds.

First we will need some simple descriptions of spatial relationships. The symbols 1 and −1 will denote distinct directions along some axis, and for some quantity Q and direction dir

Direction-Of(dir, Q)

will be true exactly when $A_s[Q]$ equals dir. The location of an object is modeled by a quantity position, and if there is no immobile object directly against an object B in direction dir we say

Free-Direction(B, dir) .

If there is an object in that direction which is directly in contact with it, say C, then we say

Contact(B, C, dir) ,

Finally, when some object C lies along direction dir from object B, we will say

Direction-Towards(B, C, dir) .

Fig. 28 contains the process specifications for motion and acceleration. The motion process occurs when a mobile object has a non-zero velocity and is free to move in the direction of its velocity (i.e., no other objects in the way). Motion has a positive influence on position of an object, in that if the velocity

```
Process Motion(B, dir)

Individuals:
  B an object, Mobile(B)
  dir a direction

Preconditions:
  Free-direction(B, dir)
  Direction-Of(dir, velocity(B))

QuantityConditions:
  A_m[velocity(B)] > ZERO

Influences:
  I+ (postiion(B), A[velocity(B)])

Process Acceleration(B, dir)

Individuals:
  B an object, Mobile(B)
  dir a direction

Preconditions:
  Free-Direction(B, dir)
  Direction-Of(dir, net-force(B))

QuantityConditions:
  A_m[net-force(B)] > ZERO

Relations:
  Let Acc be a quantity
  Acc ∝_Q+ net-force(B)
  Acc ∝_Q- mass(B)
  ; The basic QP version of F = m * a
  Correspondence((Acc ZERO)
                 (net-force(B) ZERO))

Influences:
  I+(velocity(B) A[Acc])
```

FIG. 28. Process descriptions of Newtonian motion and acceleration. In this motion vocabulary we have abstracted away the kind of motion occurring (flying, sliding, swinging, etc.) and the complexities of motion in more than one dimension. We assume sign values are assigned to directions along some axis, with magnitudes indicating distance from some arbitrarily chosen origin.

is positive the position will 'increase' and if the velocity is negative the position will 'decrease'. (The problem of mapping a quantity space onto more complex geometric frames of reference will be considered in more detail below). Acceleration occurs when a mobile object has some non-zero net force in a free direction. Acceleration provides a positive influence on velocity, and in fact the influence is qualitatively proportional to the net force and qualitatively inversely proportional to the mass of the object (the QP version of Newton's Second Law).

While this description is Newtonian, Aristotelian and impetus theories can

also be described.[19] One form of Aristotelian motion, for example, can be written as in Fig. 29. Here motion only occurs when an object is being pushed. An impetus theory is described in Fig. 30. Aristotelian theory has the problem of explaining what keeps a moving object going once it doesn't touch anything; impetus theory explains this by the push giving an object a kind of internal force or 'impetus'. While superficially like momentum, impetus kinematics is very different.[20] Impetus also differs from momentum in that it can spontaneously dissipate. Compare the dissipation of impetus with the Newtonian model of sliding friction in Fig. 31. Here friction occurs when there is surface contact, and produces a force on the object that is qualitatively proportional to the normal force and acts in a direction opposite that of the motion. The effect of friction occurs indirectly, through providing a force that changes acceleration, rather than directly as in the impetus theory.

```
Process Motion

Individuals:
  B an object, Mobile(B)
  dir a direction

Preconditions:
  Free-Direction(B, dir)
  Direction-Of(dir, net-force(B))

QuantityConditions:
  A_m[net-force(B)] > ZERO

Relations:
  Let velocity be a quantity
  velocity ∝_Q+ net-force(B)
  velocity ∝_Q- mass(B)

Influences:
  I+(position(B), A[velocity])
```

FIG. 29. Aristotelian motion. Aristotle theorized that objects required a constant push to keep them going. Note that velocity does not have an existence independent of the motion process.

Collisions are complicated in any theory of motion. The reason collisions are complicated is that they are usually described in terms of a piece of history. We will use an encapsulated history, as described in Section 3.8. The simplest description of a collision just involves a reversal of velocity, as illustrated in

[19]McCloskey [28] and Clement [50] argue that naive theories of motion in our culture correspond to impetus theories, rather than Aristotelian theories as previously suggested.

[20]In particular, impetus is not a vector quantity. Subjects vary in their beliefs as to the means of combination for impetus; they include rules like "the motion is in the direction of the biggest impetus." There are other oddities as well—for example, impetus 'remembers' not just the direction of the push but some of the previous history of directions, so that leaving a spiral tube will result in spiral motion for a little while. See [28] for details.

```
Process Motion

Individuals:
  B an object, Mobile(B)
  dir a direction

Preconditions:
  Free-Direction(B, dir)
  Direction-Of(dir, impetus(B))

QuantityConditions:
  A_m[impetus(B)] > ZERO

Relations:
  Let vel be a quantity
  vel ∝_Q+ impetus(B)

Influences:
  I+(position(B), A[vel])
```

```
Process Impart

Individuals:
  B an object, Mobile(B)
  dir a direction

Preconditions:
  Free-Direction(B, dir)
  Direction-Of(dir, net-force(B))

QuantityConditions:
  A_m[net-force(B)] > ZERO

Relations:
  Let acc be a quantity
  acc ∝_Q+ net-force(B)
  acc ∝_Q- mass(B)

Influences:
  I+(impetus(B), A[acc])
```

```
Process Dissipate

Individuals:
  B an object, Mobile(B)

QuantityConditions:
  A_m[impetus(B)] > ZERO

Relations:
  Let acc by a quantity
  A_s[acc] = A_s[impetus(B)]

Influences:
  I−(impetus(B), A[acc])
```

FIG. 30. An impetus dynamics for motion. In impetus theories of motion, a push imparts 'impetus' to an object. An object's impetus is an internalized force that keeps on pushing the object, thus causing motion. Motion eventually stops because impetus spontaneously dissipates with time.

Fig. 32. As a simplification we have assumed C is immobile so that we won't have to worry about momentum transfer between moving objects and changes of direction in more than one dimension. Even our more complicated models of collisions appear to use such encapsulated histories, such as a compound history consisting of contacting the surface, compression, expansion, and finally breaking contact. The type of collision—elastic or inelastic—that occurs could be specified by referring to a theory of materials for the objects involved.

4.3.2. *Relationship to qualitative states*

Previous work on representing motion used a *qualitative-state* representation for motion [6, 12], an abstraction of the notion of state in classical mechanics.

```
Individual View Moving-Friction

Individuals:
  B an object, Mobile(B)
  S a surface
  dir a direction

Preconditions:
  Sliding-Contact(B, S)

QuantityConditions:
  Motion(B, dir)

Relations:
  Let fr be a quantity
  fr ∝Q+ normal-force(B)
  As[fr] = −As[velocity(B)]
  fr ∈ forces-on(B)
```

FIG. 31. Moving friction in Newtonian sliding. Objects have a set forces-on, whose sum is the net force on the object. Friction is modeled by an individual view rather than a process because it contributes directly to the force on an object, rather than the derivative of the force.

```
Encapsulated-History Collide(B, C, dir)

Individuals:
  B an object, Mobile(B)
  C an object, Immobile(C)
  dir a direction
  E an event

Preconditions:
  (T contact(B, C, dir) start(E))
  (T direction-of(dir, velocity(B)) start(E))

QuantityCondition:
  (T Motion(B, dir) start(E))

Relations:
  (M A[velocity(B)] start(E)) = −(M A[velocity(B)] end(E))
  (M A[velocity(B)] during(E)) = ZERO
  duration(E) = ZERO
  (T contact(B, C, dir) end(E))
```

FIG. 32. Colliding modeled as an encapsulated history. Sometimes all that is known about a situation is the particular kind of behavior that will occur. While violating composability, encapsulated histories are the only way to evolve a history in such cases. This particular encapsulated history describes a perfectly-elastic collision with a fixed object in one dimension.

Some of the parameters that would appear in a classical description of state are represented abstractly—typically position is represented by a piece of space, and velocity by a symbolic heading. While in classical physics the type of activity is left implicit in the choice of descriptive equations, the qualitative-

state representation explicitly names the type of activity (FLY, SLIDE, etc.). Qualitative states are linked by simulation rules that map a qualitative state into the set of qualitative states that can occur next. Envisioning using such simulation rules is simple; given an initial state, use the rules to generate a set of new states. Apply the simulation rules to these states in turn, and continue in this fashion until no distinct new states are generated. Fig. 33 illustrates a physical situation and the envisionment that results. The envisionment can be used to answer simple questions directly, assimilate certain global assumptions about motion (such as assuming that a ball must or must not ever be in a particular place), and plan solutions to more complex questions. By examining the relationship between the qualitative state representation and the QP representation we will understand both more clearly.

Consider a process vocabulary comprised solely of motion and acceleration. The limit analysis for a moving object will include only the possibilities raised by dynamics, such as the acceleration due to gravity reversing the velocity of a ball or friction bringing a sliding block to a halt. The possible changes in process structure caused by kinematics—such as the ball hitting a wall or the block flying off a table—are not predicted within this vocabulary. To include them would require encoding the relevant geometry in such a way that it can be moved out of the preconditions and into the quantity conditions. To do this, we must first describe space by a *place vocabulary*[21], because we must break space up into distinct pieces that can be reasoned about symbolically. We might then try to use the entities in the place vocabulary as elements in the quantity space for position. Then the kinematic changes would be discovered by limit analysis just as the dynamical ones are.

Unfortunately, things are not so simple. First of all, we need to introduce an ordering between elements of the place vocabulary. (This ordering need not be total; we can use ambiguity in the ordering to represent our lack of knowledge about the precise heading of the moving object.) For motion in two or three dimensions this requires specifying a direction to obtain a partial order. And because we have specified a direction, we now must also specify the place we are starting from, since that will determine what the neighbors in the position quantity space are. (Consider walking out your front-door while throwing a ball up in the air. What the ball might hit changes dramatically.) However, this means the place and direction must be included in the specification of the motion process. If we could successfully add such information, an instance of the motion process in this vocabulary would begin to look like a qualitative state for the same collection of places and type of motion. The qualitative simulation rules would thus roughly correspond to a compilation of the limit analysis on this new motion vocabulary.

[21]Forbus [12] describes the principles involved and defines a place vocabulary for motion through space in a simple domain.

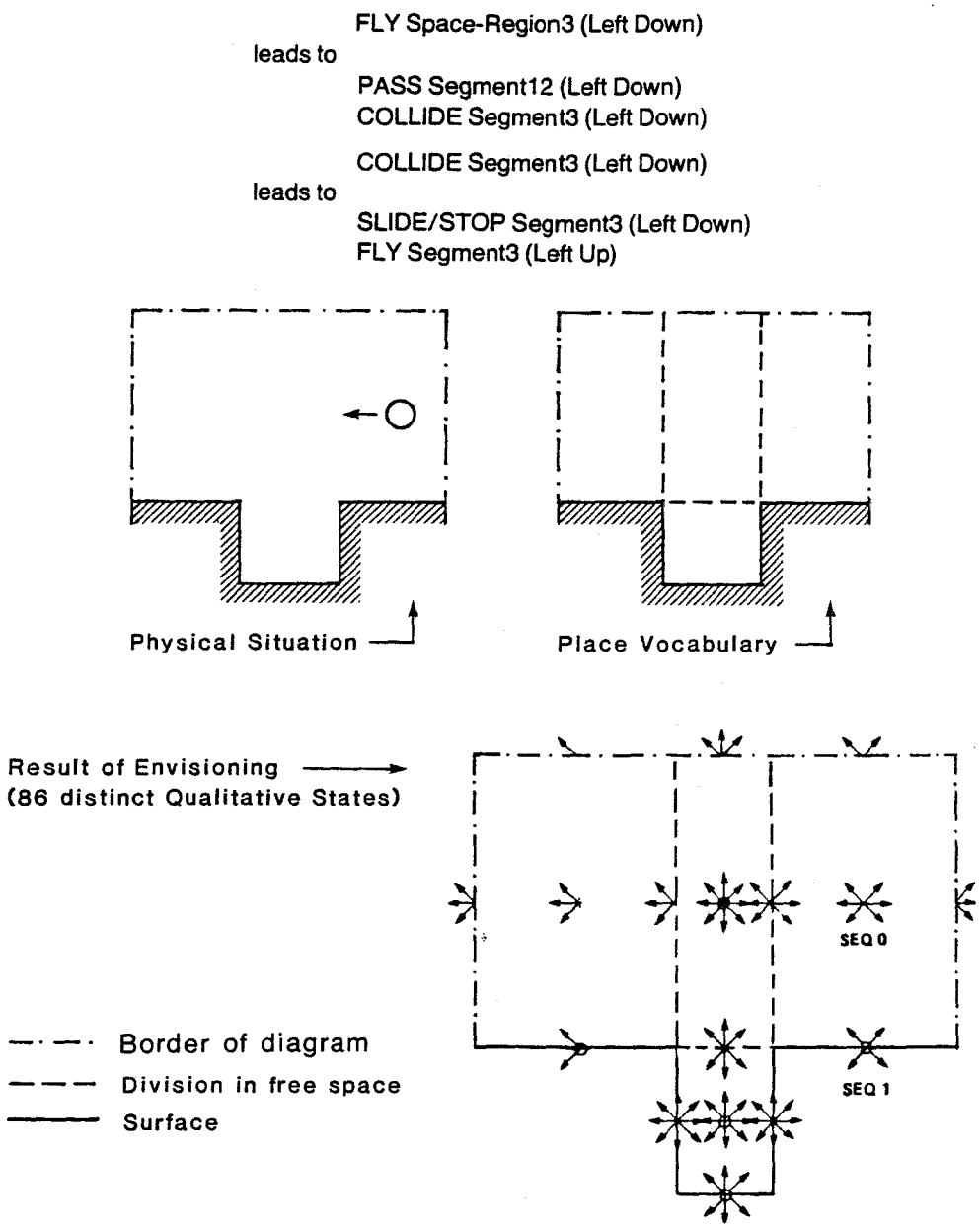


FIG. 33. Qualitative-state description of motion. Consider the ball moving leftwards. A qualitative description of space (*place vocabulary*) can be computed from the diagram and the possible ways the ball can move given that initial state are depicted schematically over the places they occur. A detailed description of two states and their relationships with other states is also shown.

From this perspective we can see the relative strengths of the two representations. For evolving descriptions of motion the qualitative-state representation seems superior, because kinematic constraints are essential to motion. However, simulation rules are an opaque form of dynamics theory—they do not contain the assumptions under which they operate. Their 'com-

piled' nature makes the qualitative-state representation inappropriate for very simple deductions (where only part of a qualitative state is known), or for more subtle analyses that involve perturbing a system. In particular, the qualitative-state representations for motion are not easily composable to form descriptions of more complex systems. The example of Section 4.5 illustrates a more subtle analysis of motion made possible by the ontology of QP theory.

4.4. Modeling materials

Let us consider what happens when we pull on something. If it doesn't move, then its internal structure is 'taking up' the force (this can happen even if it does move—try hitting an egg with a baseball bat—but we will ignore this case). Three things can happen: (1) it might do nothing (rigid behavior); (2) it might stretch (elastic behavior); or (3) it might break. For a push, (2) becomes compressed and (3) becomes crushed. We can use the notions of quantity and process provided by QP theory to state these facts. In particular, we can express the changes between these kinds of behavior by creating individual views describing these properties, introducing new elements into a quantity space for forces on an object.

The concepts involved with elasticity can be thought of in terms of applied force versus internal force. If the magnitude of the applied force is greater than that of the internal force the length of the object will change. The change in length results in an internal force that will counterbalance the applied force. Three individual views describe the states of an elastic object, either stretched, relaxed, or compressed. Fig. 34 illustrates the individual view for elastic objects and their states. To avoid the complications of shape and connectivity, we only model one-dimensional objects that have a fixed end. By convention, forces into an object (pushes) will be negative and applied forces directed outwards (pulls) will be positive.

Imagine that we apply a constant force to an elastic object (with, say, a robot hand under force, rather than position, control). An imbalance between internal and applied forces will result in the length changing. Exactly what occurs depends on the state of the elastic object (stretched, relaxed, compressed), the sign of the applied force, and the relative magnitudes of the forces (the dependence on the sign of the internal force is encoded in the state of the object via the $\propto_Q$ and correspondence). The four possibilities are stretching, compressing, and two kinds of relaxing. These processes are described in Fig. 35.

Of course, objects are not perfectly elastic. If the applied force is very small, objects will often behave rigidly. If too much force is applied an object can break or crush. The rigidity under small forces can be modeled by adding another quantity condition to stretching and compressing. For a partially-elastic object the thresholds for compressing and stretching will be called $f_{compress}$ and

```
Individual-View Elastic-Object

Individuals:
  B an object

Preconditions:
  Elastic-Substance(made-of(B))

Relations:
  Has-Quantity(B, length)
  Has-Quantity(B, Internal-force)
  Has-Quantity(B, rest-length)
  D_s[rest-length(B)] = 0
  internal-force(B) ∝_Q+ length(B)
  Correspondence((internal-force(B) ZERO)
                 (length(B) rest-length(B)))
```

```
Individual-View Relaxed

Individuals:
  B an elastic-object

QuantityConditions:
  A[length(B)] = A[rest-length(B)]
```

```
Individual-View Stretched

Individuals:
  B an elastic-object

QuantityConditions:
  A[length(B)] > A[rest-length(B)]
```

```
Individual-View Compressed

Individuals:
  B an elastic-object

QuantityConditions:
  A[length(B)] < A[rest-length(B)]
```

FIG. 34. Descriptions of elastic objects. An elastic object stores energy in reversible deformations of shape. The basic view of an elastic object relates the internal force and length, and the other three views describe the states it can be in.

$f_{stretch}$ respectively. The conditions under which crushing and breaking can be captured similarly by thresholds f_{crush} and f_{break}, which are functions of both the material and the object (to allow for dependence on the shape). The process descriptions for crushing and breaking, however, are more complex than compressing and stretching because they involve irreversible changes. This requires statements in the relations field that explicitly mention time, turning the description into an encapsulated history rather than a true process. Much of the information that must be included concerns deformations of shape and transformation of one object into several. As with kinematics, these issues are beyond the scope of QP theory.

Fig. 36 illustrates the force quantity spaces that describe different kinds of materials. In theory a taxonomy such as this one could be used for classifying a

```
process Stretching

Individuals:
  B an elastic object

Preconditions:
  ¬ Position-Constrained(B)

QuantityConditions:
  ¬ Compressed(B)
  A_s[applied-force(B)] = 1
  A_m[Applied-force(B)]
    > A_m[internal-force(B)]

Relations:
  Let SR be a quantity
  SR ∝_Q+ (A_m[applied-force(B)]
          - A_m[internal-force(B)])

Influences:
  I+(length(B), SR)
```

```
process Relaxing-Minus

Individuals:
  B an elastic object

Preconditions:
  ¬ Position-Constrained(B)

QuantityConditions:
  Stretched(B)
  A_m[applied-force(B)]
    < A_m[internal-force(B)]

Relations:
  Let SR be a quantity
  SR ∝_Q+ (A_m[applied-force(B)]
          - A_m[internal-force(B)])

Influences:
  I-(length(B), SR)
```

```
process Compressing

Individuals:
  B an elastic object

Preconditions:
  ¬ Position-Constrained(B)

QuantityConditions:
  ¬ Stretched(B)
  A_s[applied-force(B)] = -1
  A_m[applied-force(B)]
    > A_m[internal-force(B)]

Relations:
  Let SR be a quantity
  SR ∝_Q+ (A_m[applied-force(B)]
          - A_m[internal-force(B)])

Influences:
  I-(length(B), SR)
```

```
process Relaxing-Plus

Individuals:
  B an elastic object

Preconditions:
  ¬ Position-Constrained(B)

QuantityConditions:
  Compressed(B)
  A_m[applied-force(B)]
    < A_m[internal-force(B)]

Relations:
  Let SR be a quantity
  SR ∝_Q+ (A_m[applied-force(B)]
          A_m[internal-force(B)])

Influences:
  I+(length(B), SR)
```

FIG. 35. Stretching, compressing, and relaxing. The continuous changes that can occur to elastic objects when constrained by an applied force are described. The individual views of stretched, compressed, and relaxed are described in Fig. 34.

material by applying forces to it and seeing what sorts of behavior result. In a richer model of materials forces along different directions could result in different behavior (such as attempting to bend balsa wood against its grain instead of along the grain) and the effects of plastic deformation would be included.

Rigid:
⟨no processes affecting length⟩
Elastic:
⟨stretching and compressing apply⟩
Breakable:
$ZERO < f_{break}$
Crushable:
$f_{crush} < ZERO$
Partially stretchable:
$ZERO < f_{stretch}$
Partially compressible:
$f_{compress} < ZERO$
Brittle:
$f_{crush} < ZERO < f_{break}$
Partially elastic:
$f_{compress} < ZERO < f_{stretch}$
Normal:
$f_{crush} < f_{compress} < ZERO < f_{stretch} < f_{break}$

FIG. 36. Materials classified by quantity spaces. Distinct kinds of materials give rise to different quantity spaces because different combinations of processes can occur. This taxonomy should allow a material to be classified by applying forces and observing what kinds of things actually occur.

A classic AI conundrum is to be able to express in some usable form that "you can pull with a string, but not push with it" [33]. This fact can be succinctly stated, at least to a first approximation, using QP theory. First, consider what pushes and pulls are. Both concepts imply one object making contact with another to apply force. Recall that if the direction of the force is towards the object it is a push, and if the direction is away from the object then it is a pull. Obviously a push can occur with any kind of contact, but pulls cannot occur with an abutting.

Understanding how pushes and pulls are transmitted is fundamental to understanding mechanisms. For a first-pass model, consider the notion of *push-transmitters* and *pull-transmitters*. We say an object is a push-transmitter if when it is pushed, it will in turn push an object that is in contact with it, in the direction between the two contact points. Pull-transmitters can be similarly defined. This particular set of definitions is obviously inadequate for mechanisms,[22] and is only for illustration. Note also that push-transmitters and pull-transmitters need not be reflexive relations. Rigid objects are an exceptional case:

[22]Consider for example a rocker arm connected to a pivot or two blocks resting on the floor that are tied together by a length of string. In the first case a push will be transmitted in a different direction, and in the second case it can be transformed into a pull. Better theories of push- and pull-transmitters will require representing kinematics in two and three dimensions.

$\forall$ B $\in$ *object*
  rigid(B) $\Rightarrow$ ($\forall$ c_1, c_2 $\in$ contact-points(B)
    Push-Transmitter(c_1, c_2)
    $\wedge$ Push-Transmitter(c_2, c_1)
    $\wedge$ Pull-Transmitter(c_1, c_2)
    $\wedge$ Pull-Transmitter(c_2, c_1))

Strings, however, are more complicated. A string can never be a push-transmitter:

$\forall$ s $\in$ *string*
  ($\forall$ *t* $\in$ *time* (T($\neg$ Push-Transmitter(end1(s), end2(s))
    $\wedge$ $\neg$ Push-Transmitter(end2(s), end1(s))) t))

But if it is taut it can be a pull-transmitter:

$\forall$ s $\in$ *string*
  ($\forall$ t $\in$ *time*
    (T Taut(s) t) $\Rightarrow$ (T Pull-Transmitter(end1(s), end2(s)) t)
    $\wedge$ (T Pull-Transmitter(end2(s) end1(s)) t))

Now the problem becomes how to define Taut. As a first pass, let ends-distance be a type of quantity representing the distance between the ends of the string. Then we can define Taut as an individual view:

Individual-View Taut

Individuals:
  s a string

QuantityConditions:
  $\neg$ A_m[ends-distance(s)] $<$ A_m[length(s)]

This model assumes that only the ends of the string contact other objects—it would fail for a rope hanging over a pulley, for instance. A better model is to divide up the string into segments according to whether or not that part of the string is in contact with a surface. A string is then taut if each segment that is not in contact with a surface is taut:

$\forall$ S $\in$ *string*
  ($\forall$ seg $\in$ segments(geometry(S)) Free-Segment(seg, S) $\Rightarrow$ Taut(seg)) $\Rightarrow$
    Taut(S)

This of course ignores the fact that the non-free segments may not be tight, as say for string lying on the floor. A full definition would also require tension along the entire string, but we have strayed far enough from dynamics already.

4.5. An oscillator

Dynamical reasoning involves more than just simulation. By analyzing the possible behaviors of a situation we can produce a summary of its behavior and eventual disposition (e.g., [12]). In classical physics these analyses are often concerned with stability. Here we will examine a simple situation involving motion and materials, ascertain that it oscillates, and perturb it to figure out under what conditions it will remain stable.

Consider the block B connected to the spring S in Fig. 37. We will model the spring S as device satisfying Hooke's Law (see Fig. 34). Initially we will assume the spring cannot break. To model the position constraint imposed on the spring's length by being rigidly connected to the block, and to set the origin of position to the location at which the spring is relaxed, we assume:

length(S) $\propto_{Q+}$ position(B)
Correspondence(length(S) rest-length(S)) (position(B) ZERO))

Suppose the block is pulled back so that the spring is extended. Initially, we also assume that the contact between the block and the floor is frictionless. What will happen?

Since initially the spring is stretched (i.e., A[length(S)] > A[rest-length(S)]), the spring will exert a force. This will in turn exert a force on the block which, since the block is free to move leftwards (S is not immobile), will cause an

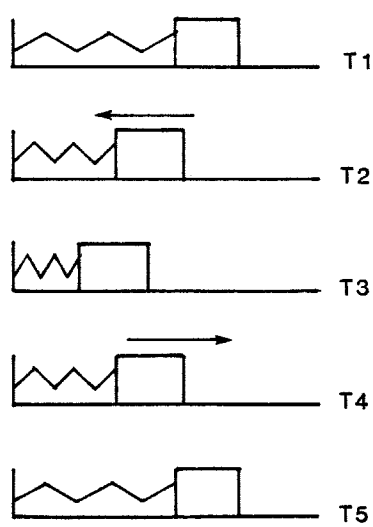


FIG. 37. A sliding block. Here is a system we will analyze to determine what it does and how different factors, such as whether or not there is friction, affect its behavior.

acceleration. So the initial view and process structures are:

VS: {Stretched(S)}
PS: {Acceleration(B, −1)}.

However, A[velocity(B)] will change from ZERO in an instant (by case (1) of the equality change law), so the process structure will become

PS: {Acceleration(B, −1), Motion(B, −1)}.

Since D_s[position(B)] = −1, by the $\varkappa_{Q+}$ above we have D_s[length(S)] = −1 as well, and by the correspondence in the definition of elastic objects, D_s[internal-force(S)] = −1 as well. The next limit point is reached when A[length(S)] = A[rest-length(S)], making S relaxed instead of stretched. When this occurs A[net-force(B)] = ZERO, thus ending the acceleration. The motion, however, continues. The process and view structures become:

VS: {Relaxed(S)}
PS: {Motion(B, −1)}.

This state of affairs will last but an instant, since position is still decreasing. As the position moves past ZERO we will have

VS: {Compressed(S)}
PS: {Motion(B, −1), Acceleration(B, 1)}.

The only limit point that can be reached occurs in the quantity space for B's velocity, i.e., A[velocity(B)] = ZERO. When that occurs the motion will stop, leaving:

VS: {Compressed(S)}
PS: {Acceleration(B, 1)}

Since Acceleration directly influences velocity, this state of affairs will instantly change to:

VS: {Compressed(S)}
PS: {Motion(B, 1), Acceleration(B, 1)}.

The conclusion that the next change results in

VS: {Relaxed(S)}
PS: {Motion(B, 1)}

with an instantaneous change to

VS: {Stretched(S)}
PS: {Motion(B, 1), Acceleration(B, −1)}

which lasts for an interval and then yields

VS: {Stretched(S)}
PS: {Acceleration(B, −1)}

follows in the same way. However, this situation matches the initial situation—the quantity spaces, view and process structures, and D_s-values are the same. Thus we can conclude that an oscillation is occurring. Note that the view and process structures must be the same, because in principle the preconditions might have changed.

Some of the assumptions made in producing the process history can now be perturbed to examine the effects of different physical models. For instance, suppose the spring is crushable and breakable, as defined previously. Then there are limit points around rest-length(S) that correspond to the occurrence of crushing or breaking. It seems crushing must be ruled out by assumption, since the machinery we have developed so far does not allow us to rule it out via contradiction. We can however deduce that the spring won't break under the conditions above.

If we can prove that the block will go out no further than when it started then we can claim that it won't break because it didn't break in the first place. This requires an energy argument. The energy theory we will use is very simple. There are certain types of quantities that are *energy-quantities*, which are qualitatively proportional to certain other quantities and exist whenever they do. Two kinds of energy are kinetic energy and 'spring' energy. For every object there is a total energy, which is the sum of its energy quantities (Fig. 38 describes systems and energy quantities more formally, and Fig. 39 describes sources, sinks, and conservation laws).

Here the system is the mass and spring combination. At time t1, the block is still but the spring is streched, i.e.,

(M A[Velocity(B)] t1) = ZERO
(M A[length(S)] t1) > A[rest-length(S)]

which means that

(M total-energy(SYSTEM) t1) > ZERO .

If energy is conserved and there is no influx of energy, then we know

∀ t ∈ *time*
After(t, t1) ⇒
¬ (M total-energy(SYSTEM) t) > (M total-energy(SYSTEM) t1)

```
Energy-Quantity(kinetic-energy)

; velocity gives rise to kinetic energy

∀ B ∈ object
  Has-Quantity(B, velocity) ⇒
    (Has-Quantity(B, kinetic-energy)
     ∧ kinetic-energy(B) ∝Q+ m[velocity(B)]
     ∧ (A[kinetic-energy(B)] = ZERO
        ↔ A[velocity(B)] = ZERO))

Energy-Quantity(spring-energy)

; an internal force gives rise to "spring" energy

∀ B ∈ object
  Has-Quantity(B, internal-force) ⇒
    (Has-Quantity(B, spring-energy)
     ∧ spring-energy(B) ∝Q+ m[internal-force(B)]
     ∧ (A[spring-energy(B)] = ZERO
        ↔ A[internal-force(B)] = ZERO))

; the total energy of an object is the sum of its energy quantities

∀ B ∈ object
  Has-Quantity(B, total-energy) ∧ Set(energy-quantities(B))
  ∧ ∀ q ∈ quantities(B) Energy-Quantity(q) ⇒ q(B) ∈ energy-quantities(B)
  ∧ total-energy(B) = sum-over(energy-quantities(B))

; the energy of a system is the sum of the energy in its objects

∀ sys ∈ system Set(objects(sys)) ∧ (∀ b ∈ objects(sys) Physob(B) ∨ System(B))
  ∧ Has-Quantity(sys, total-energy) ∧ Set(energy-quantities(sys))
  ∧ (∀ B ∈ objects(sys)
       Subset(energy-quantities(B), energy-quantities(sys)))
  ; ignore converse case (assume all members must be from some part)
  ∧ total-energy(sys) = sum-over(energy-quantities(sys))
```

FIG. 38. A simple energy theory—energy and systems. The predicate Energy-Quantity asserts that its argument is a quantity representing a kind of energy. Energy quantities occur as a consequence of objects having particular types of quantities. The energy of a system is the sum of the energy quantities for its parts.

This means that the block can only go out as far as it was at t1, since if it ever went out farther we would contradict the previous statement.

4.5.1. *Stability analysis*

To further analyze this system, we must treat the processes that occur as a compound process. We can start by writing an encapsulated history, including properties of the objects taken over the piece of history (a cycle of the oscillator) so defined. We want to perform an energy analysis, so we will include the total energy of the system (total-energy(SYSTEM)) and the maximum length of the spring over a cycle (max-length(S)), since length(S) gives us an

```
; processes can be sources and sinks w.r.t. a system

∀ pi ∈ process-instance ∀ sys ∈ system ∀ q ∈ quantity-type
   Source(pi, sys, q) ≡
   (∃ B ∈ objects(sys) Influences(pi, q(B), +1))
   ∧ ¬ (∃ B ∈ objects(sys) Influences(pi, Q(B), −1))
; define sinks similarly, ignore cross-flows

∀ pi ∈ process-instance ∀ sys ∈ system
   Energy-Source(pi, sys) ≡
   (∀ q ∈ quantity Energy-Quantity(q) ∧ Source(pi, sys, q))
   ∧ (∀ q ∈ quantity Energy-Quantity(q) ⇒ ¬ Sink(pi, sys, q))
; energy sinks are defined analogously

; simple form of conservation:

; if you don't kick it it won't get any higher. .

∀ sys ∈ system ∀ i ∈ time
   (∀ pi ∈ process-instance
      Energy-source(pi, sys) ⇒
         (∀ I1 ∈ during(i) (T Status(pi, INACTIVE) I1))) ⇒
         ¬ (M total-energy(sys) end(i)) > (M total-energy(sys) start(i))

; more complex version:

∀ sys ∈ system ∀ i ∈ interval
   (∀ pi ∈ process-instance
      Energy-Source(pi, sys) ⇒
         (∀ I1 ∈ during(i) (T Status(pi, INACTIVE) I1)))
   ∧ (∀ pi ∈ process-instance
         Energy-Sink(pi, sys) ⇒
            (∀ I1 ∈ during(i) (T Status(pi, INACTIVE) I1))) ⇒
            (M A[total-energy(sys)] start(i)) = (M A[total-energy(sys)] end(i))
```

FIG. 39. A simple energy theory—sources, sinks, and conservation. There are several forms of energy conservation, some stronger than others. The weakest says that if there is no inflow then the energy never increases. The strongest says that in a closed system the energy is always the same.

indication of 'spring energy'. We assume the relations for the compound process include:

```
max-length(S) ∝Q+ total-energy(SYSTEM)
correspondence((max-length(S), ZERO), (total-energy(SYSTEM) ZERO))
```

since during each cycle there will be some time during which all of the energy is in the spring. To perform an energy analysis we must re-write any inequalities in the quantity conditions in terms of energy, to wit:

```
QuantityConditions:
   A[total-energy(SYSTEM)] > ZERO .
```

Thus if the total energy of the system ever reaches ZERO during an occurrence of the compound process it will no longer be active, because the total energy of the system is zero only when the spring is relaxed and the block is unmoving. Note that the quantity condition is no longer tied to a particular episode of the encapsulated history. This means that, unlike the encapsulated histories previously encountered, we cannot use this one for simulation. Instead, we use it to analyze global properties of the system's behavior.

We can use the basic QP deductions on this new description to determine the consequences of perturbing the model of the situation in various ways. Each perturbation is represented by a process that influences one of the parameters that determines the energy of the system. For example, suppose friction were introduced into the system. Its effect will be modeled by introducing a new quantity, e-loss(SYSTEM), the energy lost during a cycle. Then

$$D_s[\text{total-energy(SYSTEM)}] = -1 ,$$

and we can deduce, via limit analysis, that the quantity condition above will eventually be false, and so the oscillation will eventually stop. Suppose the system is pumped so that its energy is increasing (i.e., $D_s[\text{total-energy(SYSTEM)}] = 1$). Then while the quantity condition above will remain true, the energy will be continually increasing, which means the force on the spring will increase over time (since during part of the cycle the energy is all in the spring, and the spring energy is qualitatively proportional to the internal force of the spring). If the spring is breakable, then there will be a limit point in the quantity space for the spring's force that will eventually be reached. So the spring could break if the system is frictionless and pumped.

Let us examine in detail what happens if the oscillator is subject to friction, but we pump it with some fixed amount of energy per cycle, as would happen in a mechanism such as a clock. Is such a system stable? We will call the energy lost to friction over a cycle e-loss(SYSTEM) and the energy added to the system over a cycle e-pump(SYSTEM). The only things we will assume about the friction process in the system is that

Relations:
  e-loss(SYSTEM) $\propto_{Q+}$ total-energy(SYSTEM)
  correspondence((e-loss(SYSTEM), ZERO), (total-energy(SYSTEM), ZERO))

Influences:
  I−(total-energy(SYSTEM), A[e-loss(SYSTEM)])

The loss being qualitatively proportional to the energy is based on the fact that the energy lost by friction is proportional to the distance traveled, which in turn is proportional to the maximum length of the spring, which itself is qualitatively proportional to the energy of the system, as stated above.

The lower bound for the energy of the system is ZERO, and an upper bound for energy is implicit in the possibility of the parts breaking. The result, via the $\propto_Q$-statement above, is a set of limit points on the quantity space for e-pump(SYSTEM). If we assume e-pump(SYSTEM) is within these limit points then there will be a value for total-energy(SYSTEM), call it e-stable(SYSTEM), such that:

$$\forall\, t \in \mathit{cycle}$$
$$(M\ A[\text{total-energy(SYSTEM)}]\ t) = (M\ A[\text{e-stable(SYSTEM)}]\ t)) \Rightarrow$$
$$(M\ A[\text{e-loss(SYSTEM)}]\ t) = (M\ A[\text{e-pump(SYSTEM)}]\ t)$$

Note that e-stable(SYSTEM) is unique because $\propto_Q$ is monotonic. If the energy of the system is at this point, the influences of friction and pumping will cancel and the system will stay at this energy. Suppose

$$(M\ A[\text{total-energy(SYSTEM)}]\ t) > (M\ A[\text{e-stable(SYSTEM)}]\ t)$$

over some cycle. Then because the loss is qualitatively proportional to the energy, the energy loss will be greater than the energy gained by pumping, i.e., D_s[total-energy(SYSTEM)] = −1, and the energy will drop until it reaches e-stable(SYSTEM). Similarly, if total-energy(SYSTEM) is less than e-stable(SYSTEM) the influence of friction on the energy will be less than that of the pumping, thus D_s[total-energy(SYSTEM)] = 1. This will continue until the energy of the system is again equal to e-stable(SYSTEM). Therefore for any particular pumping energy there will be a stable oscillation point. This result is actually a qualitative version of the proof of the existence and stability of limit cycles in the solution of a differential equation. It is surprising just how little information about the system we needed to draw these conclusions, and it will be interesting to see what other results from the classical theory of differential equations can be derived from qualitative information alone.

5. Further Consequences

Qualitative process theory provides a representational framework for a certain class of deductions about the physical world. In this section we examine the consequences of this framework for several 'higher-level' issues in common-sense physical reasoning. Several of these issues arise in reasoning about designed systems, while others arise more generally.

5.1. Distinguishing oscillation from stutter

We have seen that envisioning—generating all the possible behaviors of a system—can be performed by repeated limit analysis. The result is a graph of situations which can be traversed to form any of the possible histories for the objects involved. In walking this graph we may find a terminal state (either

because the situation is quiescent, because we do not know how to evolve a history past a certain kind of event or because we simply haven't bothered) or we might find a loop. A loop must be summarized if we are to get a finite description of the system's behavior. There are several ways to produce such summaries. In some domains the major regularity is spatial, in which case we would produce descriptions like "the ball is bouncing around inside the well" [12]. Another type of concise summary is possible when a system is oscillating, since there is a pattern of activity that occurs over and over again.

While oscillation in the physical system results in loops in the envisionment, there are other circumstances that give rise to loops as well. In part this is due to the qualitative nature of the descriptions used. Consider the situation illustrated in Fig. 40. Initially there are two flows, one from A to B and the other from B to C. What can happen? Limit analysis reveals three alternatives, corresponding to each of the flows stopping individually and to both ending simultaneously (see Fig. 41). In the cases where one flow stops before the other, the flow that continues will decrease the amount, and hence pressure, of the water in container B so the other flow will start again. These cycles of activity do not correspond to physical oscillations; they are an artifact of our qualitative physics. A better description of this behavior is that the change in level 'follows' the other change. In other words, we have a decaying equilibrium. We will call the behavior represented by these degenerate cycles *stutter*. How can we distinguish stutter from true oscillation?

Physically an oscillation requires that the system have some form of inertia or hysteresis. This means that, at least for some part of the system, when the cause of the change stops acting, the change will continue for a while afterwards. A real oscillation will therefore include process episodes that last over an interval, whereas stutter—a kind of 'mythical oscillation—will only include process episodes that last but an instant.

Fortunately the equality change law (Section 3.6.4) provides a way of distinguishing these cases. In Fig. 41, for example, the transitions marked with an 'i' take place in an instant. Therefore we have two instances of stutter, corresponding to the two fluids participating in a decaying equilibrium.

A similar phenomena occurred in the boiler model presented earlier (Section 5.2). Fig. 42 depicts the envisionment. Notice that if t-boil(WATER) rises faster than temperature(WATER) (due to the increasing pressure), the boiling will

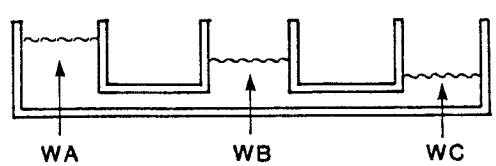


FIG. 40. Three container example. Suppose we have three containers partially filled with water and connected by pipes, as shown. If we assume the water moves slowly, what can happen?

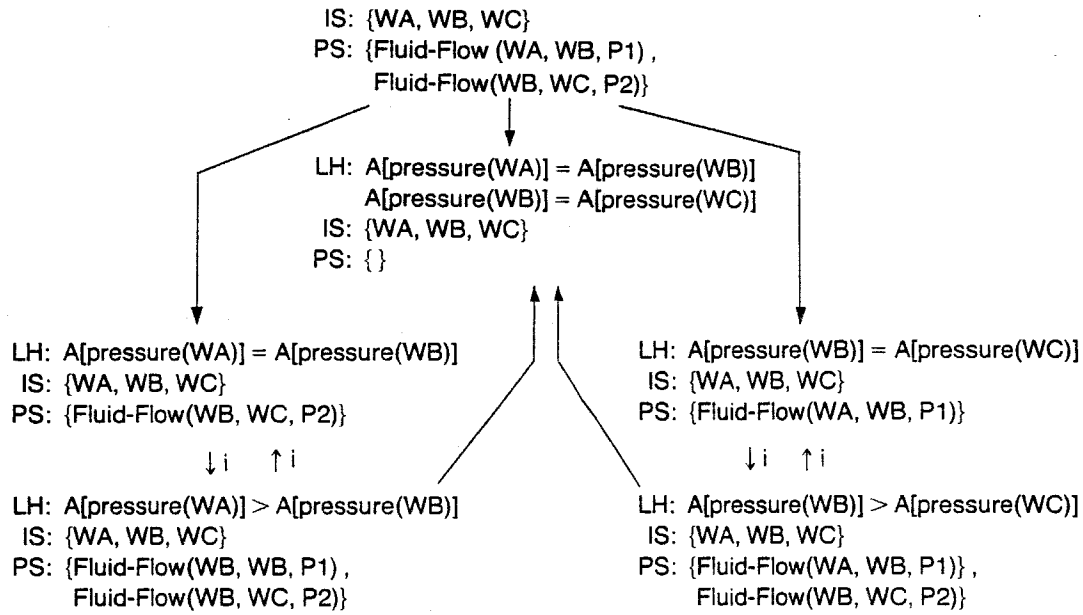


FIG. 41. Stutter in fluid flow. This graph of transitions between process structures formed by repeated limit analysis contains two cycles, neither of which correspond to physical oscillations. For simplicity, we ignore the possibility of the contained liquids vanishing as a result of the flows.

stop. Since this change in the inequality relationship between the quantities is a change from equality, by case (1) of the equality change law it will occur in an instant. This in turn means that t-boil(WATER) was only influenced for an instant. When the boiling stops only the heat flow is acting, so temperature(WATER) will rise, and thus by case (2) of the equality change law the return to equality will occur in an instant. Therefore this cycle is an instance of stutter as well, corresponding to a rising equilibrium.

Being able to distinguish stutter from oscillation means we can write rules that summarize the process history. For example, when stutter occurs we can note the D_s-values for the quantities involved and assert that one kind of change is 'following' another, a decaying or rising equilibrium. Informal observations suggest that novices in a domain often confuse stutter and oscillation, and even experts who describe the situation as decaying or rising equilibrium are able to reconstruct the view of stutter as an oscillation. These informal observations need to be examined in the light of empirical data, but if true it may be useful in testing QP theory as a psychological model.

5.2. Causal reasoning

We use causality to impose order upon the world. When we think that "A causes B", we believe that if we want B to happen we should bring about A, and that if we see B happening then A might be the reason for it. Causal

FIG. 42. Stutter in the boiler model. The temperature and pressure will be continuously increasing in the boiler, but unless the changes in the links marked 'i' are recognized as occurring in an instant, the system will appear to be oscillating.

reasoning is especially important for understanding physical systems, as noted in [7, 39]. Exactly what underlies our notion of causation in physical systems is still something of a mystery.

Consider the representations used in physics. Typically equations are used to express constraints that hold between physical parameters. A salient feature of

equations is that they can be used in several different ways. For example, if we know "X = A + B", then if we have A and B we can compute X, but also if we have X and A we can compute B. It has been noted that in causal reasoning people do *not* use equations in all possible ways [10, 40]. Only certain directions of information flow intuitively correspond to causal changes. I propose the following *causal directedness* hypothesis:

Causal directedness hypothesis. *Changes in physical situations which are perceived as causal are due to our interpretation of them as corresponding either to direct changes caused by processes or propagation of those direct effects through functional dependencies.*

This section will attempt to justify that hypothesis.

First, I propose that causality requires some notion of mechanism.[23] Imagine an abstract rectangle of a particular length and width. If we imagine a rectangle that is longer, it will have greater area. There is no sense of causality in the change from one to the other. If however we imagine the rectangle to be made of some elastic material and we bring about the increased length by stretching it, then we are comfortable with saying "the increased length causes the area to increase". Qualitative process theory asserts that processes are the mechanisms that directly cause changes. The quantities that can be directly influenced by processes are in some sense independent parameters, because they are what can be directly affected. All other quantities are dependent, in that to affect them some independent parameter or set of independent parameters must be changed. This suggests representing the relationships between parameters for causal reasoning in terms of functions rather than constraint relations.

Some examples will make this clearer, as well as emphasizing that the point is not academic. In generating explanations of physical systems, it is often useful to characterize how the system responds to some kind of change (this variety of qualitative perturbation analysis was invented by De Kleer, who calls it *incremental qualitative analysis* (IQ)). One way to perform IQ analysis is to model the system by a constraint network, in which the relationships are modeled by 'devices' that contain local rules that enforce the desired semantics.[24] The values of quantities are modeled by the sign of their change—

[23]In its most general form, this proposal is not new (Bunge [4] surveys various proposals concerning the nature of causality). For example, Heise [23] proposes *operators* as a mechanism that underlies all causal relations. The proposal presented here is more specific.

[24]These examples are drawn from systems implemented in CONLAN [14], a constraint language. The graphical notation for constraint networks is similar to logic diagrams, except that 'terminals' are given explicit names and the 'devices' are multi-functional. The technique described here is actually a simplication of De Kleer's algorithm. However, the models in [7] sometimes used directional rules rather than constraint laws, although no criteria was provided for selecting which direction in a constraint law is appropriate for causal reasoning.

increasing, decreasing, or constant. To perform an analysis, a value corresponding to a hypothesized change is placed in a cell of the constraint network representing the system. The rules associated with the constraint network are then used to deduce new values for other cells in the network. The propagation of information models the propagation of changes in the system, with dependency relationships between the cell values corresponding to causal connections. For example, if the value of cell A was used to deduce the value of cell B, we would interpret this as "The change in A caused the change in B". Fig. 43 illustrates fragments from two different models.[25] The top fragment states that heat is the product of the temperature of the 'stuff' and the amount of 'stuff', and the bottom fragment is the definition of sodium concentration in a solution.

In building a causal argument it is possible to reach an impasse—a quantity receives a value, but no further values can be computed unless an assumption is made. The safest assumption is that, unless you know otherwise, a quantity doesn't change. The problem lies in determining which quantity to make the assumption about. Suppose we assume that the amount of stuff is constant. Then we would conclude that an increase in heat causes an increase in temperature, which makes sense. However, suppose we assume instead that the temperature remains constant. We are left with the conclusion that an increase in heat *causes* the amount of stuff to decrease! Barring state changes, this does not

Correct causal argument: "The increasing heat causes the temperature to rise"
Incorrect causal argument: "The increasing heat causes the amount of fluid to rise"

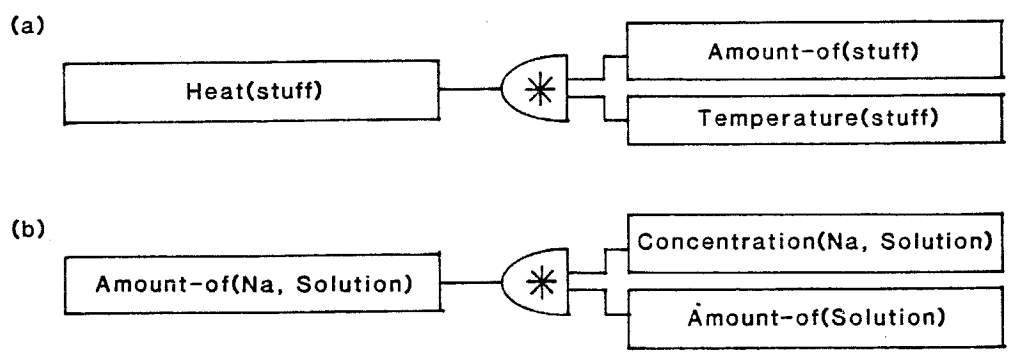


FIG. 43. Constraint representation of relationships. In the constraint networks, the boxes (cells) denote quantities. The relationship between the parameters is expressed by a multiplier constraint connecting them. Box (a) is drawn from the model for a piece of 'stuff' used in an effort to represent a student's understanding of heat exchangers. Box (b) is drawn from a model of a kidney to be used in explaining the syndrome of inappropriate secretion of anti-diuretic hormone (SIADH).

[25]These model fragments are drawn from an attempt to implement the model of a student's understanding of a heat exchanger described in [49], and an early version of the kidney model described in [2].

correspond to our notion of causality. In the second fragment the problem is more serious—increasing sodium will cause the amount of water to increase, *if* the rest of the kidney is working as it should! To do this requires a complicated feedback mechanism that is triggered by detecting an increased sodium concentration, not by the definition of concentration itself.

The problem lies in the ontological impoverishment of the constraint representation. Temperature and concentration are not directly influenced by processes (at least in most people's naive physics). Physically, they are *dependent* variables, and thus are not proper subjects of assumptions when constructing causal arguments. Amount of stuff, on the other hand, can be directly affected, so assuming it does not change will avoid generating ill-formed causal arguments. Fig. 44 illustrates.

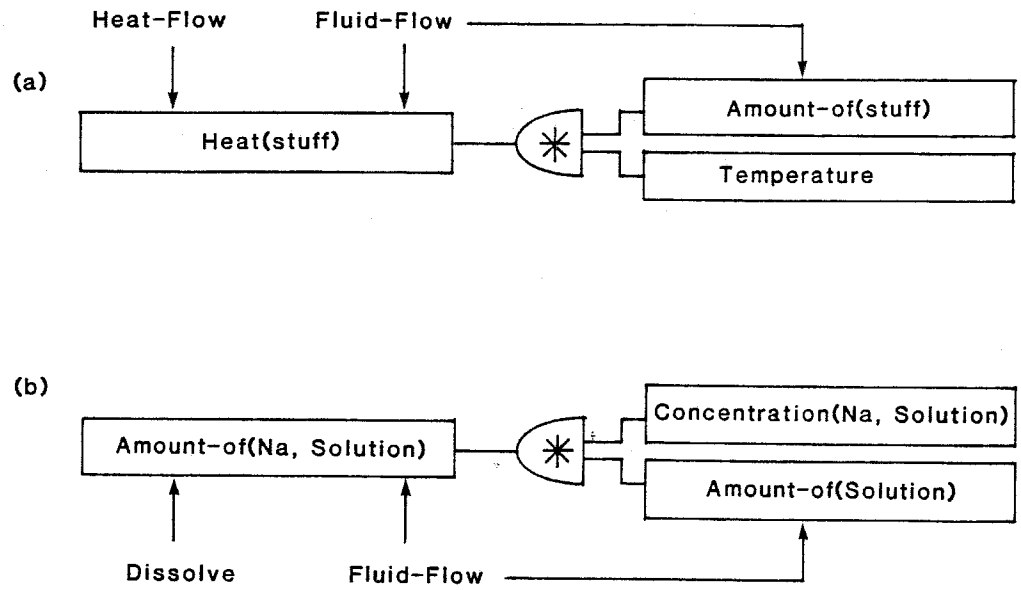


FIG. 44. Model fragments with possible processes. Models from Fig. 43 with the quantities annotated with the (likely) processes that might affect them. Note that certain quantities (temperature, concentration) cannot be directly changed. These are *dependent* quantities, and should not be the subject of assumptions in building causal arguments.

Of course, the proper assumptions to make really concern which processes are active and how infiuences are resolved. If we do not represent processes, we can only assume facts about quantities. If we assume a quantity is constant and later discover that assumption is wrong, we are left in the dark about *why* that assumption was wrong. For example, if the amount of stuff turns out not to be constant, we can look for fluid flows or state changes to explain why it isn't. Since processes tend to have more than one effect, there is some chance that the contradiction can lead to discovering more about the system rather than just being a bad guess.

5.3. Qualitative proportionalities revisited

The previous section argued that functional dependence is central to the kind of 'incremental' causality that people find useful in reasoning about the physical world. As discussed earlier, one goal of naive physics should be to develop a theory of observation. One use of observation is to interpret measurements in terms of theories [18], but another role for observation is in developing the physical theories themselves. While this problem has been studied before (c.f. [27]), the target representations have been equations. As a result the learning procedure has relied on numerical data and cannot build theories around weaker information. Learning constraint laws also differs from learning causal connections. As noted previously, an equation carries only part of what we know about a domain. Construction a learning theory for physical domains will require ways to learn process descriptions and causal connections.

One way to learn about a system is to 'poke' it and see what it does. The observed behavior can be used to make conjectures about causal connections between the parts of the system, and further experiments can be made to test the conjectures. This requires some notation to express the local causal connections conjectured on the basis of these simple observations. This requirement helped motivate the definition of $\propto_Q$ (see Section 2), which asserts that a functional dependence exists between two quantities. If whenever we increase parameter A in a system we observe that parameter B increases, the result can be expressed as:

$$B \propto_{Q+} A,$$

A physical explanation for the dependence comes from writing the $\propto_Q$ within the scope of an individual view or process.

More powerful statements about a system or domain will require extensions of $\propto_Q$. To see what is involved, consider the analogous problem of learning how an old-fashioned typewriter works.[26] If the space bar is pushed, the carriage will move to the left. This is analogous to the kind of statement that can be made with $\propto_Q$. But lots of other things can happen to move the carriage, namely all of the letter keys and a few more. Thus it would be useful to be able to state that we know *all* of the influences (at least, within the current grasp of the situation) on some particular parameter. Suppose also that we just wanted to move the paper up without changing anything else. The return bar would move the paper up, but before doing so would return the carriage to the right. Being able to say there are no (known) intervening parameters is then also a useful ability.

To see how these notions can be expressed, consider the collection of $\propto_Q$ relations that hold at some instant in time. For any quantity, the $\propto_Q$-statements

[26]This is not proposed as a serious example because the quantity definitions and $\propto_Q$ would apply only in some very abstract sense.

relevant to it can be thought of as a tree with the dependent quantity at the root and the 'independent' quantities at the leaves. A plus or minus denotes the sense of the connection (whether or not it will reverse the sign of the change in the input). Thus

$$Q_0 \propto_Q Q_1$$

only specifies that Q_0 is on some branch 'above' Q_1.

Fig. 45 illustrates such a dependency tree. Suppose we are trying to cause Q_0 to change. If we don't want to change Q_2, then Q_3 or Q_1 are our only choices. We need a way to express that (at least within our knowledge of the situation) there are no intervening parameters. To say this, we use

$$\propto_Q\text{-immediate}(Q_0, Q_1)$$

which can be modified by + or − as before. $\propto_Q$-immediate adds a single link to the tree of dependencies. Another problem is to find all the ways to bring a change about, or to prove that changing one thing won't cause a change in some other quantity of interest. We do this by stating that a particular collection of quantities together 'closes off' the tree—there will be exactly one quantity for each branch. Out notation will be

$$\propto_Q\text{-all}(\langle\text{quantity}\rangle, \langle\text{plus-set}\rangle, \langle\text{minus-set}\rangle)$$

which means that there is a function which determines the quantity, relies on the quantities in the two sets solely, and is strictly increasing in its dependence on the quantities in the plus-set and strictly decreasing in its dependence on the quantities in the minus-set. If a quantity is not mentioned in a $\propto_Q$-all statement, then either it is irrelevant to the quantity of interest, it depends on some quantity in the $\propto_Q$-all statement (above the slice of the tree it represents), or some quantity in the $\propto_Q$-all statement depends on it. By ruling out the other two possibilities, independence can be established.

As a rule $\propto_Q$-statements will not hold for all time. In the typewriter analogy, imagine the carriage at the end of its travel—hitting the space bar will no longer result in movement. More to the point, consider Q_0 given by:

$$Q_0 = (a - b * Q_2) * Q_1 .$$

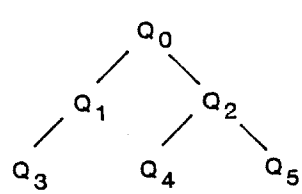


FIG. 45. A tree of functional dependencies.

Note that:

$$\begin{aligned} \text{if } a > b * Q_2, &\quad Q_0 \propto_{Q+} Q_1 \\ a = b * Q_2, &\quad \neg\, Q_0 \propto_{Q-} Q_1 \\ a < b * Q_2, &\quad Q_0 \propto_{Q-} Q_1 \end{aligned}$$

In the case of equality, Q_0 and Q_1 are not related at all, and in the other two cases the sign of the function connecting them is different. Thus the collection of $\propto_Q$-statements that are true for a system can vary as a function of the values of the quantities, which is why they usually appear within some individual view or process. The idea of a *mode* of a system in 'real' physics roughly corresponds to particular process and view structures which hold during the system's operation.

5.4. Differential qualitative analysis

The idea of a comparison in IQ analysis suggests a complementary qualitative reasoning technique. IQ analysis concerns the relationship between two situations, one of which is the direct result of things happening in the other. Another case of interest concerns situations that are just slightly different from one another—an 'alternate-world' interpretation. For instance, we often have an idea of the different consequences that would result if something were changing a bit faster—if we put the heat up on the stove the water in the kettle would boil sooner, and if our arm were quicker the serve would have been returned. Such inferences are essential in debugging and monitoring execution of plans that involve physical action, and performing sensitivity analyses to evaluate a proposed design. The language needed to express such conclusions is in part the same as that used in IQ analysis—amounts are either the same, increased, decreased, or indeterminate as compared with the old situation. Answering these kinds of questions will be called *differential qualitative* analysis.

Let us consider a situation S_1. If we can get a new situation S_1 by changing a single ordering in S_1 or by changing the status of a single process or view instance in S_1, we will call S_2 an *alternate* of S_1. There are two kinds of changes that may occur as a result of perturbing S_1. First, other quantities can change. Second, the process history for the situation itself may change, apart from any changes made to define S_2 in the first place. For example, punching a hole in the bottom of a kettle could let all the water drain out before it boils. Even changes in orderings can lead to further changes in the histories of the individuals involved—e.g., if we reduce the intensity of a flame but still turn it off in five minutes, boiling may be prevented.

Let $DQ(q, S_1, S_2)$ for some quantity q be the sign of the difference between it in two alternate situations S_1 and S_2. Then the inequality order between the

defines DQ-values, as follows:

$$(M\ q\ S_1) > (M\ q\ S_2) \leftrightarrow DQ(q, S_1, S_2) = -1$$
$$(M\ q\ S_1) < (M\ q\ S_2) \leftrightarrow DQ(q, S_1, S_2) = 1$$
$$(M\ q\ S_1) = (M\ q\ S_2) \leftrightarrow DQ(q, S_1, S_2) = 0$$

Since situations can occur over intervals, the inequality orderings for instants must be extended. For equality this is simple:

$$\forall\ q_1, q_2 \in \textit{quantity}\ \forall\ i \in \textit{time}$$
$$(M\ q_1\ i) = (M\ q_2\ i) \equiv \forall\ i_1 \in \text{during}(i)\ (M\ q_1\ i_1) = (M\ q_2\ i_1)$$

For the other cases the choice is less clear. The strongest version of greater-than is having it hold over every instant in the interval:

$$\forall q_1, q_2 \in \textit{quantity}, i \in \textit{interval}$$
$$(M\ q_1\ i) > (M\ q_2\ i) \equiv$$
$$(\forall\ i_1 \in \text{during}(i)\ (M\ q_1\ i_1) > (M\ q_2\ i_1))$$

however, the following will also suffice:

$$\forall q_1, q_2 \in \textit{quantity}, i \in \textit{interval}$$
$$(M\ q_1\ i) > (M\ q_2\ i) \equiv$$
$$[(\exists\ i_1 \in \text{during}(i)\ (M\ q_1\ i_1) > (M\ q_2\ i_1))$$
$$\wedge\ (\forall\ i_1 \in \text{during}(i)\ \neg (M\ q_1\ i_1) < (M\ q_2\ i_1))]$$

A version of $<$ for intervals may be similarly defined.

An episode in a parameter history has several numbers associated with it. The relationships between these numbers allows new DQ-values to be determined. The first number is *rate*, e.g., the D_m of the quantity the parameter history is about. The second number is the duration of the interval associated with the episode. The third number is the difference in the value measured at the beginning and end of the interval, which we will call the *distance*.

How are these numbers related? Intuitively we know that if the rate changes, the duration of time will vary inversely, or the distance the value moves will vary accordingly for the same duration. Implicit in this simple intuition is the assumption that the rate is constant during the interval, i.e., that the function defining the change of the quantity is linear and time invariant. This often is not the case, so we must require that either the beginning or the end of the two episodes being compared are the same. If we apply DQ analysis only to alternate situations as defined above this restriction will be satisfied.

With these assumptions, the DQ-value of the distance is just the product of the DQ-values of the rate and duration. Thus we can draw conclusions such as

"if the rate is higher then over the same time the distance traveled will be greater" and "if the duration is shorter and the rate is the same then the distance traveled will be less." These inferences are illustrated in Fig. 46.

The direct historical consequences of these changes can be characterized by their effects on limit analysis. Consider a collection of limit hypotheses for a p-component. Recall that each hypothesis concerns a possible change in the process structure, brought about by changes in quantities that cause changes in quantity conditions. Suppose a particular limit hypothesis is chosen as representing what actually occurs. This means the change it stands for happens before the changes represented by the other hypotheses. If in an alternate situation this hypothesis has an increased duration (a DQ-value of 1) or one of the other limit hypotheses has a decreased duration (a DQ-value of −1), then in fact a different change could occur. Once again, the weak nature of our information prevents us from actually deciding if a different change would occur—but we at least know that it is possible in these circumstances.

; definition of distance

$\forall S \in \textit{episode}$ distance(Q) = (M A[Q] end(S)) − (M A[Q] start(S))

Suppose we have alternate situations S_1 and S_2, with a quantity Q in both of them.

$DQ[\text{distance}(Q, S), S_1, S_2] = DQ[\text{rate}(Q), S_1, S_2] * DQ[\text{duration}(Q), S_1, S_2]$

Then assuming $\text{time}(\text{start}(S_1)) = \text{time}(\text{start}(S_2))$.

$$DQ[\text{rate}(Q), S_1, S_2] = 1 \wedge DQ[\text{duration}(Q), S_1, S_2] = 0 \Rightarrow DQ[\text{distance}(Q), S_1, S_2] = 1$$

$$DQ[\text{rate}(Q), S_1, S_2] = -1 \wedge DQ[\text{distance}(Q), S_1, S_2] = 0 \Rightarrow DQ[\text{duration}(Q), S_1, S_2] = 1$$

i.e., "If Q is going faster then it will get farther in the same time" and "If Q is going slower it will take longer to go the same distance."

FIG. 46. Differential qualitative analysis. Differential qualitative analysis answers quentions about how a situation would change if parts of it are perturbed.

6. Discussion

This paper has described qualitative process theory, which attempts to model aspects of commonsense reasoning about physical domains. To summarize:

(1) Our theories about how things change in the physical world have a common character. Physical processes are the mechanisms by which change occurs. Reasoning about processes—their effects and limits—form an important part of our commonsense physical reasoning.

(2) Numerical values can be usefully represented by the quantity space, which describes the value of a number in terms of inequalities. The quantity space is an appropriate representation because processes usually start and stop when order relationships between particular quantities change.

(3) QP theory provides the means to draw several types of basic, qualitative, deductions, including describing what is happening in a physical situation (finding view and process structures), reasoning about the combined effects of several processes (resolving influences), and predicting when processes will start and stop (limit analysis). These basic deductions can be woven together to perform more complex inferential tasks, such as envisioning.

(4) QP theory can be used to model several interesting physical phenomena for commonsense reasoning, including flows, state changes, motion, materials, energy, changing equilibria, and oscillation.

(5) QP theory provides a highly constrained account of physical causality (all changes are due to a finite vocabulary of processes) and a useful notation for expressing causal connections between quantities ($\propto_Q$).

(6) QP theory provides a structured role for the use of experiential and default knowledge in physical reasoning—for example, in resolving influences and choosing or ruling out alternatives in limit analysis.

(7) QP theory partially specifies a language for writing qualitative dynamics theories. In particular, the primitives are simple processes and individual views, the means of combination are sequentiality and shared parameters, and the means of abstraction are naming these combinations, including encapsulating a piece of a history.

6.1. Application areas

While designed to be a theory about naive physics, qualitative process theory has other potential applications. Two are discussed below.

6.1.1. *ICAI and engineering problem solving*

Since many engineered devices are implemented as physical systems, QP theory should be useful in reasoning about them. Perhaps the most immediately feasible application is providing part of a representation language for intelligent computer-aided instruction (ICAI). An important part of expert's knowledge of a system is a qualitative understanding of how it works. To the extent that QP theory models our qualitative understanding of dynamical systems, a program using it can generate explanations that a student will find easy to understand. Indeed, QP theory was developed in part to be used in the STEAMER project, whose goal is to provide instruction about steam propulsion plants for Navy trainees.[27]

[27]The STEAMER project is a joint enterprise of the Navy Personnel Research and Development Center and Bolt, Beranek, and Newman, Inc. See [46] for an overview.

One interesting implementation strategy is to construct a *tutor compiler*. Current qualitative-reasoning programs work from first principles in constructing explanations. This is analogous to setting a human domain expert down in front of a system he has never seen before and expecting him to understand it well enough to generate coherent explanations in real time. Using human instructors to teach this way is obviously a bad idea, so why should we expect our programs to do better? One alternative is to construct a program which takes as its input a system to be understood and a specification of the class of questions which are to be asked about it. The output of this program would be a specialized tutoring program that could, in real time, answer that class of questions about the system in question. This technique would have several advantages, for example, the qualitative-reasoning system in the compiler itself need not be especially fast, and more sophisticated techniques for generating explanations could be exployed than would otherwise be possible (such as McDonald's MUMBLE [32].

As extension theories are developed, QP theory should become useful in other kinds of engineering and control tasks as well. Individual views could be used to express desirable and undesirable operational characteristics. For example, in operating a boiler the fuel-air ratio must not become too rich or too lean; in either case smoke pours out the boiler stack, which is bad if you want not to be seen and combustion efficiency, hence fuel economy, will drop. A good boiler design will provide operating regions in which the individual views representing these undesired conditions are inactive. Similarly, these descriptions could be used in synthesizing control strategies, by determining what measurements indicate a state from which a view instance representing an undersirable condition will become active and what corrective action must be taken to ensure that the particular change will not occur.

Another interesting possibility is building a *hypothesizer*. A hypothesizer is an interpretation module which either takes measurements from operators of a system or gathers data itself from instruments, and will evaluate an operator's theories about what is happening in the system. Such a program could serve as a devil's advocate, pointing out inconsistencies in an operator's theory or suggesting alternate interpretations for the data. This would be useful because it seems that a common source of human error in operating complex systems (such as nuclear power plants) is premature commitment to a particular theory about the state of the system (see [37]). For example, the incident at the Three Mile Island reactor might not have happened if the operators had thought of the alternate explanation for the overpressure in the reactor vessel—that instead of being too high, the level of cooling water was too low, thus causing a boiling that raised the pressure.

6.1.2. *Economic modeling and decision support systems*

Historically, the success of differential equations in physics led to attempts to

apply them to problems from other fields, such as economics. To the extent that differential equations prove useful in reasoning about a domain, QP theory should be similarly useful. In economics, for example, physical limitations often prove important. Storage capacities, transportation capacities, and rates of manufacture are important examples (see [20, 45]).[28] The features which make qualitative models useful for physical reasoning, such as the ability to characterize the classes of things that can happen even with very little data, should be useful in other domains, especially in domains where numerical data is unreliable or hard to come by.

However, caution seems advisable in attempting such applications. There seems to be no real agreement on what mathematical descriptions are appropriate in economics, hence it will be hard to judge whether a qualitative model is correct. In addition, the very structure of the domain can change with time; for instance, the tax code can change. These factors make modeling economics much harder than modeling physics.

6.2. Other work

The first attempts to formalize processes modeled them as collections of interacting automata [3] or extended STRIPS-like operators [24]. Let us examine each in turn.

Brown's automata-based system was designed to generate explanations for intelligent computer-aided instruction. Quantities and processes were represented by individual automata whose states represented classes of values or activities (such as a quantity decreasing or a particular activity in a sequence occurring). Time was modeled by specifying that automata representing quantities changed instantly while automata representing processes took an interval of time to change. Although arbitrary LISP code was permitted in specifying state transitions, in practice state changes were predicted on state changes in other automata. While adequate for generating explanations of fixed phenomena, the automata representation is too brittle for most reasoning tasks. For example, there is no influence-like mechanism for dynamically combining effects, thus all interactions must be foreseen in advance by the model builder. The process models are similar to encapsulated histories, in that they presuppose the outcome of the activity of the processes they describe. Hence such models will be insensitive to changing conditions.

Hendrix's system was designed to provide a world model for robot planning. While a significant advance over the models of action available at the time, the importance of qualitative descriptions had not yet been understood. For example, all quantities were real numbers, and relationships between

[28]Interestingly, Samuelson was one of the first to describe the possibility of using qualitative models and to point out that their inherent ambiguity would make prediction difficult. Subsequent developments in qualitative modeling, however, suggest his views were overly pessimistic, at least for physical domains.

parameters were expressed as numerical constraint equations. The process descriptions were used in simulation, solving simultaneous equations in the process descriptions to determine when the collection of active processes would change. Since the goal was to model general processes (non-physical as well as physical), add lists and delete lists were also used to specify effects. Qualitative process theory, by using qualitative descriptions and focusing on physical processes only, can be used in several other kinds of deductions in addition to simulation, often requiring less information to draw interesting conclusions.

Recently several attempts have been made to model temporal reasoning by Allen [1] and McDermott [30]. Allen's model is the one assumed here, both because meet seems to be the appropriate relationship between pieces of a history and because modeling instants as 'very short' intervals makes formalizing certain facts involving derivatives easier. McDermott's axioms for time contain several interesting ideas, including the chronicle representation of possible futures and its implications for planning. Unfortunately, McDermott expects too much of the temporal logic. For example, the logic includes the notion of a 'lifetime', how long you can assume a fact to be true once you have observed it to be true. McDermott claims lifetimes must be provided outside the logic, by fiat ("The senses actually tell you about persistences"), because having axioms that provide persistences can lead to contradictions. This ad hoc notion is needed precisely because the logic is developed independently from a theory of dynamics. Given a dynamics (and the ability to make closed-world assumptions about individuals and relationships), we can *deduce* what will and will not change. If we need an estimate of how long something will remain true, we can figure out how long it is likely to be before something that can change it occurs. To use McDermott's example, if you look at a boulder you might be able to estimate that if you came back in 50 years it would still be there (a weaker conclusion than implied by the notion of lifetimes, but it will do). However, if you are told that there is dynamite underneath, your estimate will be considerably different. In either case, if you came back the next day and discovered the boulder was some distance from its original location, you would have some theory about why, not just the feeling that your senses had lied to you. In addition, McDermott's model of quantities uses average rate instead of derivatives, which means many of the dynamical conclusions described here (such as distinguishing oscillation from stutter) cannot be drawn.

6.3. Current directions

Since the original publication of qualitative process theory, several projects have adopted or extended some of its ideas. In particular:

(1) Ben Kuipers has adopted a subset of QP theory for analyzing protocols of causal reasoning in medicine, including an implementation of rules to reason about changes within a process structure [25, 26].

(2) Reid Simmons has developed process representations for geologic inter-

pretation by qualitative simulation, including the use of a diagram. His system extends the quantity-space representation by using quantitative information, including representing values by intervals and using specific equations to describe functional dependencies [42].

(3) Johan De Kleer and John Seely Brown have extended their device-centered qualitative physics for machines to include inequality information so that state transitions can be more precisely modeled (see this volume). Also, Brian Williams has developed a similar device-centered physics for reasoning about VLSI circuits, focusing on the interrelationships between intuitive and formal mathematical models and the importance of continuity (also in this volume).

(4) Al Stevens, Dan Weld, and Albert Boulanger are using qualitative process theory in constructing a theory of explanations for machines [47].

(5) Alan Collins and Dedre Gentner are using qualitative process theory to express theories of evaporation in order to understand how to shift from one level of description to another. Also, we are using QP theory in developing a psychological theory of learning for physical domains [19].

ACKNOWLEDGMENT

Mike Brady, John Seely Brown, Alan Collins, Johan De Kleer, Dedre Gentner, Pat Hayes, David McAllester, Drew McDermott, Bruce Roberts, Reid Simmons, Al Stevens, Gerald Sussman, Dan Weld, Brian Williams, and Patrick Winston have all influenced the development of this theory in various ways. The bulk of this research was carried out at the MIT Artificial Intelligence Laboratory, under support from the Defense Advanced Research Projects Agency. Portions of this work were supported by the Navy Personnel Research and Development Center as part of the STEAMER project at Bolt, Beranek, and Newman, Inc.

REFERENCES

1. Allen, J., Maintaining knowledge about temporal intervals, TR-86, Computer Science Department, University of Rochester, Rochester, NY, 1981.
2. Asbell, I., A constraint representation and explanation facility for renal physiology, MIT SM Thesis, Cambridge, MA, 1982.
3. Brown, J.S., Burton, R.R. and Zdybel, F., A model-driven question-answering system for mixed-initiative computer-assisted instruction, *IEEE Trans. Systems Man Cybernet.* **3** (3) (1973).
4. Bunge, M., *Causality and Modern Science* (Dover, New York, 1979).
5. Collins, A., Warnock, E., Aiello, N. and Miller, M., Reasoning from incomplete knowledge, in: D. Bobrow and A. Collins (Eds.), *Representation and Understanding* (Academic Press, New York, 1975).
6. De Kleer, J., Qualitative and quantitative knowledge in classical mechanics, TR-352, MIT AI Lab, Cambridge, MA, 1975.
7. De Kleer, J., Causal and teleological reasoning in circuit recognition, TR-529, MIT AI Lab, Cambridge, MA, 1979.
8. De Kleer, J. and Brown, J., Assumptions and ambiguities in mechanistic mental models in: D. Gentner and A. Stevens (Eds.), *Mental Models* (Erlbaum, Hillsdale, NJ, 1983).

9. De Kleer, J. and Sussman, G., Propagation of constraints applied to circuit synthesis, MIT AI Lab Memo No. 485, Cambridge, MA, 1978.
10. diSessa, A., Phenomenology and the evolution of intuition, in: D. Gentner and A. Stevens (Eds.), *Mental Models* (Erlbaum, Hillsdale, NJ, 1983).
11. Fikes, R. and Nillson, N., STRIPS: A new approach to the application of theorem proving to problem solving, *Artificial Intelligence* **2** (1971) 189–208.
12. Forbus, K., A study of qualitative and geometric knowledge in reasoning about motion, TR-615, MIT AI Lab, Cambridge, MA, 1981.
13. Forbus, K. and Stevens, A., Using qualitative simulation to generate explanations, BBN Tech. Rept. No. 4490, prepared for Navy Personnel Research and Development Center, 1981; also in: *Proceedings Third Annual Conference of the Cognitive Science Society*, August, 1981.
14. Forbus, K., A CONLAN primer, BBN Tech. Rept. No. 4491, prepared for Navy Personnel Research and Development Center, 1981.
15. Forbus, K., Qualitative reasoning about physical processes, in: *Proceedings Seventh International Joint Conference on Artificial Intelligence*, Vancouver, BC, August, 1981.
16. Forbus, K., Qualitative process theory, MIT AI Lab Memo No. 664, Cambridge, MA, 1982; revised 1983.
17. Forbus, K., Qualitative reasoning about space and motion in: D. Gentner and A. Stevens (Eds.), *Mental Models* (Erlbaum, Hillsdale, NJ, 1983).
18. Forbus, K., Measurement interpretation in qualitative process theory, in: *Proceedings Eighth International Joint Conference on Artificial Intelligence*, Karlsruhe, Germany, August, 1983.
19. Forbus, K. and Gentner, D., Learning physical domains: Towards a theoretical framework, in: *Proceedings Second International Machine Learning Workshop*, June, 1983.
20. Forrester, J.W., *Principles of Systems* (MIT Press, Cambridge, MA, 1968).
21. Hayes, P.J., The naive physics manifesto, in: D. Michie (Ed.), *Expert Systems in the Micro-Electronic Age* (Edinburgh University Press, Edinburgh, 1979).
22. Hayes, P.J., Naive physics 1 – Ontology for Liquids, Memo, Centre pour les Études Semantiques et Cognitives, Geneva, Switzerland, 1979.
23. Heise, D.R., *Causal Analysis* (Wiley, New York, 1975).
24. Hendrix, G., Modeling simultaneous actions and continuous processes, *Artificial Intelligence* **4** (1973) 145–180.
25. Kuipers, B., Getting the envisionment right, in: *Proceedings National Conference on Artificial Intelligence*, Pittsburgh, PA, 1982.
26. Kuipers, B. and Kassirer, J., How to discover a knowledge representation for causal reasoning by studying an expert physician in: *Proceedings Eighth International Joint Conference on Artificial Intelligence*, Karlsruhe, Germany, August, 1983.
27. Langley, P., Rediscovering physics with BACON.3, in: *Proceedings Sixth International Joint Conference on Artificial Intelligence*, Tokyo, Japan, August, 1979.
28. McClosky, M., Naive theories of motion, in: D. Gentner and A. Stevens (Eds.), *Mental Models* (Erlbaum, Hillsdale, NJ, 1983).
29. McCarthy, J. and Hayes, P.J., Some philosophical problems from the standpoint of artificial intelligence, in: B. Mettzer and D. Michie (Eds.), *Machine Intelligence* **4** (Edinburgh University Press, Edinburgh, 1969).
30. McDermott, D., A temporal logic for reasoning about processes and plans, *Cognitive Sci.* **6** (2) (1982).
31. McDermott, D. and Doyle, J., Non-monotonic logic I, *Artificial Intelligence* **13** (1980) 41–72.
32. McDonald, D., Natural language generation as a computational problem: an introduction, in: J.M. Brady and R.C. Berwick (Eds.), *Computational Models of Discourse* (MIT Press, Cambridge, MA, 1983).
33. Minsky, M., Personal communication.

34. Minsky, M., A framework for representing knowledge, MIT AI Lab Memo No. 306, Cambridge, MA, 1974.
35. Moore, R., Reasoning from incomplete knowledge in a procedural deduction system, MIT AI Lab, TR-347, Cambridge, MA, 1975.
36. Moore, R., Reasoning about knowledge and action, MIT Ph.D. Thesis, Cambridge, MA, 1979.
37. Pew, R., Miller, D. and Feeher, C., Evaluation of proposed control room improvements through analysis of critical operator decisions, Electric Power Research Institute Rept. NP-1982, 1982.
38. Reiter, R., A logic for default reasoning, *Artificial Intelligence* **13** (1980) 81–132.
39. 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, 1977.
40. Riley, M., Bee, N. and Mokwa, J., Representations in early learning: The acquisition of problem solving strategies in basic electricity/electronics, University of Pittsburgh Learning Research and Development Center, Pittsburgh, PA, 1981.
41. Shearer, J., Murphy, A. and Richardson, H., *Introduction to System Dynamics* (Addison-Wesley, Reading, MA, 1967).
42. Simmons, R., Representing and reasoning about change in geologic interpretation, TR-749, MIT AI Lab, Cambridge, MA, 1983.
43. Stallman, R. and Sussman, G., Forward reasoning and dependency-directed backtracking in a system for computer-aided circuit analysis, *Artificial Intelligence* **9** (1977) 135–196.
44. Stanfill, C., The decomposition of a large domain: Reasoning about machines, in: *Proceedings National Conference on Artificial Intelligence*, Washington, DC, August, 1983.
45. Stansfield, J., Conclusions from the commodity expert project, MIT AI Lab Memo 601, Cambridge, MA, 1980.
46. Stevens, A. et al., STEAMER: Advanced computer aided instruction in propulsion engineering, BBN Tech. Rept. No. 4702, Cambridge, MA, 1981.
47. Weld, D., Explaining complex engineered devices, BBN Tech. Rept. No. 5489, Cambridge, MA, 1984.
48. Williams, B., Qualitative analysis of MOS circuits, *Artificial Intelligence* **24** (1984) this volume.
49. Williams, M., Hollan, J. and Stevens, A., Human reasoning about a simple physical system, in: D. Gentner and A. Stevens (Eds.), *Mental Models* (Erlbaum, Hillsdale, NJ, 1983). (Eds.),
50. Clement, J., A conceptual model discussed by Gallileo and used intuitively by physics students, in: D. Gentner and A. Stevens (Eds.), *Mental Models* (Erlbaum, Hillsdale, NJ, 1983).

Received March 1982; revised version received July 1983