MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY

Working Paper No. 218 July 1981

Proposal For a Study of Commonsense Physical Reasoning

Kenneth D. Forbus

Abstract

Our common sense views of physics are the first coin in our intellectual capital; understanding precisely what they contain could be very important both for understanding ourselves and for making machines more like us. This proposal describes a domain that has been designed for studying reasoning about constrained motion and describes my theories about performing such reasoning. The issues examined include qualitative reasoning about shape and physical processes, as well as ways of using knowledge about motion other than "envisioning". Being a proposal, the treatment of these issues is necessarily cursory and incomplete.

This report describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the laboratory's artificial intelligence research is provided in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research Contract number N00014-80-C-0505.

CONTENTS

FIGURES

1. Introduction

1.1 Statement of the Problem

Much of the early life of a human being is spent learning about the physical world. Infants play with blocks, toys, and other objects, learning in the process about shapes, materials, and how things move. Our common sense views of physics are the first coin in our intellectual capital; understanding precisely what they contain could be very important both for understanding ourselves and for making machines more like us. I propose to study how we reason about simple physical situations involving shape and motion, leading up to a qualitative understanding of mechanisms such as clocks. To state the question more precisely:

What is required to perform common sense reasoning about simple physical situations involving motion constrained by shape?

By constrained motion I mean either motion constrained by a surface or by a rotational connection. A block sliding on a surface or a rolling ball are examples of motion constrained by a surface, and a swinging pendulum or a turning gear are examples of constrained rotational motion. Motion without some kind of constant contact, such as a ball bouncing around a room or shooting dice, will not be considered.

The rest of this section outlines the theories I have developed about this kind of reasoning and some specific examples from the domain. The next section describes the relation of this proposal to other work in Artificial Intelligence. After that the domain to be reasoned about is given a precise characterization and the issues involved in reasoning about it are explored in more detail. Finally a plan of attack and tentative schedule are given.

1.2 Statement of the Theories

As an overview, here is a summary of the theories as they stand at present:

1. Reasoning about constrained motion requires reasoning about shapes. The shape of an object determines what kind of surface it presents to the world, and its placement determines what other surfaces it will come in contact with. For common sense reasoning involving shapes I conjecture that

placement information should be exact and shape information should be expressed qualitatively. While many inferences appear to require precise information about surface normals, I believe that most common sense inferences concerning motion can be performed just using rough notions of curvature and continuity of surface normals. I believe the appropriate way to deal with the additional complexities of having a third dimension is to reduce the problem to a set of two dimensional problems. Features of the problems suggest projections (or slices) within which useful results can be obtained, and these results are then combined by "lifting" them back into the 3D representation.

2. A qualitative theory of processes plays a central role in common sense physical reasoning. Physical processes describe the influences on the parameters of objects that make them change through time. Reasoning about the limits of rigidity in the face of applied force can capture some intuitions about how materials bend and break. The process of resolving forces in classical mechanics has as its analog in Naive Physics the process of determining which influence will dominate the change in some parameter. This process is performed by ignoring certain of the influences and ordering the rest. Knowing what may safely be ignored and the relative magnitudes of influences for particular cases is probably a major part of the physical knowledge we gain from experience apart from the basic laws. Simple deductions involving functional dependence, such as the effect of friction on the time and distance an object will move, also appear to rely on an understanding of process.

3. Qualitative reasoning about physical systems, and especially in understanding motion, has mainly been orgainized around the notion of envisioning. Envisioning is a form of qualitative simulation that generates all possible states of a system. In many cases envisioning is neither practical nor desirable. Simpler methods should be used to answer simpler questions. In addition, many questions about motion are only indirectly concerned with qualitative states of motion, so that little or no explict searching need occur. A better reprentation for qualitative knowledge about motion than simulation rules must be used to gain flexiblity.

1.3 Examples

The test of these theories will be a program or programs. The proposed mode of operation is as follows:

A situation is given to the program, which consists of a collection of objects whose shapes and placements are specified, along with connections between them. Various initial conditions, such as impulsive or constant forces, motion in some direction, and states of energy sources (springs, etc.) can be postulated as well. Questions can be asked about the basic situation (kinematics and stability), about the effects of the initial conditions, and about global properties of any motion that results, such as oscillations, possible collisions, and final fate. Questions that require exact quantitative data (such as required for incremental time simulation) are not allowed.

Below are several examples of the kind of reasoning which are to be captured. Italics indicate the imagined response of the program. While the use of English for communication is intended to remain a fantasy, the use of diagrams is not.

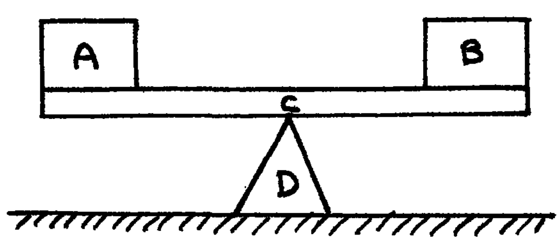
1.3.1 Example One

Figure 1 illustrates the problem.

->Suppose when it is released, C tips to the left.

Fig. 1. Balance problem

At first the assembly is held still. Suppose when it is released C tips to the left.



Unless something else is pushing the A side down or the B side up, A must be more dense than B. This assumes C has uniform density. It also relies on the fact that D appears to be in the middle and the sizes of A and B appear to be the same.

->Could both A and B go downward?

Yes, if C bends or breaks.

->Could both A and B go up?

Yes, if somebody lifted C.

->What if C is pushed rightward?

C will tip rightward and down.

->What if A is picked up?

B will push C down and to the right.

->And if A and B are picked up at the same time?

C will remain where it is.

->What if both are pushed down?

Nothing will happen, unless C bends or breaks.

1.3.2 Example Two

Figure 2 illustrates a situation in deKleer's IJCAI paper. The pivot without a name, the stick A and the ball B form a pendulum, and a fixed position pin lies in its path.

->How far to the left could B get?

P1, because in that position A will hit C.

->If it was released where it is now, would it reach that position?

Yes, unless the friction of the pivot prevents it.

->Is there a way in which B could get to X in the same motion?

Perhaps, if A were a string instead of a stick.

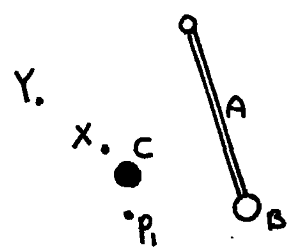
->Could it reach Y moving leftwards?

No, because A is too short even if it were a string.

->Is it possible to get to Y at all?

Fig. 2. The Pin and the Pendulum

A is initially assumed to be a stick, free to rotate about a pivot at one end and with a ball B on the other end. C is a pin with a fixed position.



Yes, if A is a string and it is moving to the right and over the pivot.

->How might A break?

Something external might break it, B could be very heavy, or B could be moving fast and cause A to break when it hits C.

->Suppose A does break when it hits C. How will the distance B travels to the left depend on the velocity when it breaks?

The faster it is travelling, the farther it will go.

1.3.3 Example Three

Figure 3 shows the physical layout. In the initial situation nothing is moving. The change being considered is the removal of the stick A.

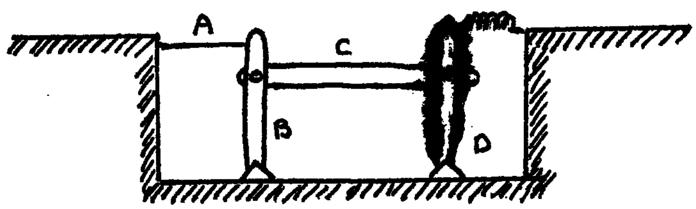
->Explain what might happen.

The rotational connections between B, C and D will cause them to move as a unit. This will cease to be true if something breaks and will be less true if they are flexible.

Gravity is pulling all the parts down, and the position of B and D indicates they would fall to the left should they be free to move. The three factors that could be preventing motion are the spring, the stick,

Fig. 3. Spring-Linkage problem

In the begining nothing is moving.



---

and the friction in the rotational connections.

If the friction is very high, removing both the spring and the stick will cause no change in the position of the assembly. Otherwise it can be ignored, since the effect friction has on the start of motion is much larger than the end, and the assembly cannot move very far anyway.

Ignoring friction, the state of the spring determines what might happen. If the spring is neutral, the assembly would just remain where it was, or possibly move leftward if the weight of the assembly was enough to stretch the spring. If the spring is compressed the assembly would move to the left, and if expanded would move to the right. The foregoing assumes that the spring is strong enough to push or pull the weight of the assembly.

->What if A was a string instead of a stick?

The analysis would be the same, except that the spring could not be compressed and have the strength to move the assembly. If such were the case, the assembly would already be in motion to the left because a string here could only restrict motion to the right.

1.3.4 Example Four

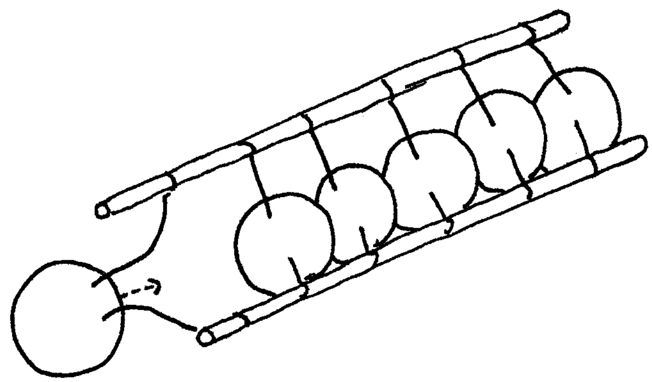
Figure 4 illustrates the initial situation, with ball A about to be released.

->What happens when A is released?

It will swing down and collide with B. The axis of the collision is aligned with the axis of the other

Fig. 4. Executive Toy

The six balls are in line and touching. This gadget has been sold as a conversation piece.



---

balls, so momentum will travel down them until the last one. Since nothing stops it from moving, it will swing out as a continuation of the motion of A. If there is friction it won't go quite as high as A originally was.

->What will the situation look like eventually?

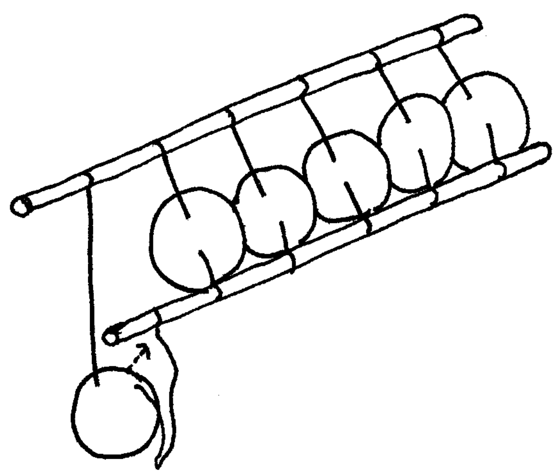
Assuming friction, F and A will gradually stop moving and all balls will be hanging straight down.

->What if A starts out in this position (Figure 5) instead?

A will swing down and collide with B, but the axis of collision is not aligned with the axis of the balls. This means they will scatter around in various ways moving both parallel and perpendicular to the original axis they formed. Assuming friction, eventually they will settle as before.

Fig. 5. Executive Toy Revisited

The heavy arrow shows the direction of initial motion. The program's conclusion about the motion of the parts should be that they all move in various directions. It is important to draw conclusions like this (and about the eventual state of a system) without the use of simulation.



1.3.5 Example Five

The situation consists of a cylinder with a longitudinal slice taken out of it, shown in cross section by Figure 6. Initially it is held at rest.

->If it is released, what will happen?

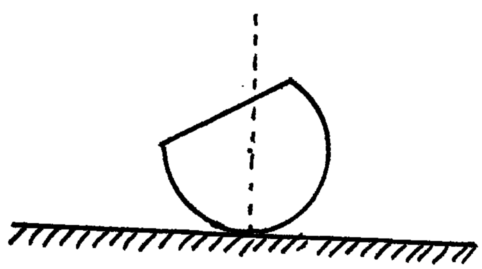
If it was just a cylinder then nothing would happen. But there is some stuff missing, and more of it is missing on the left (as determined by drawing a line down the center of the circle) than on the right. This means the center of gravity is somewhere on the right, and so it will roll to the right.

->Where will it be eventually?

The eventual location is unknowable without some estimate of the friction. However, the orientation will be such that the slice line is horizontal (which makes the center of gravity over the geometric

Fig. 6. Shape effects Rolling

Symmetry appears to be an important part of our thinking. The shape representation should emphasize symmetries and allow the deduction of the consequences of asymmetry.



---

center of the object).

->Is there any other way it might end up, assuming it rolls?

Yes, if it rolled enough the flat side would be on the table, and there it would stop.

1.3.6 Example Six

Figure 7 illustrates a four-bar linkage, with link D grounded.

->Can D move?

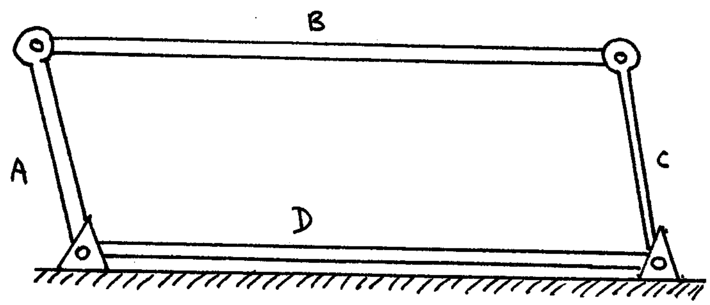
Only if one or both pivots are moved.

->What parts can move independently?

ABC will move as a unit.

Fig. 7. Four Bar Linkage

While quantitative information is to be used in the spatial representation, the goal is not to make quantitative conclusions about motion.



->What if I push C to the left?

A and B will also move to the left, A around the grounded pivot.

1.3.7 Example Seven

The sliding toy analyzed in [24] is shown in Figure 8.

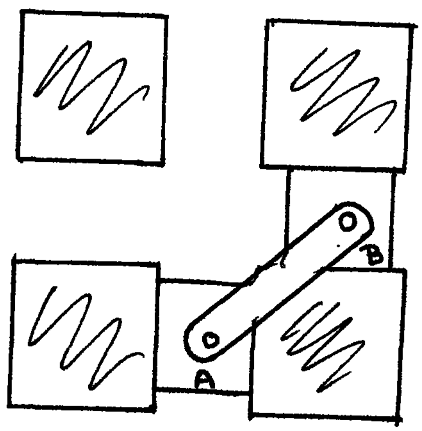
->What if A is pushed up?

It will move up and B will move out. This will change when the pivot point of A is at the center of the intersection of the channels. Past this point B will move in again, until the pivot point of B is itself at the center point. Then it will stop.

->Why?

Because B cannot go down A's channel.

Fig. 8. Sliding Toy
The arrow indicates the direction of the proposed force.



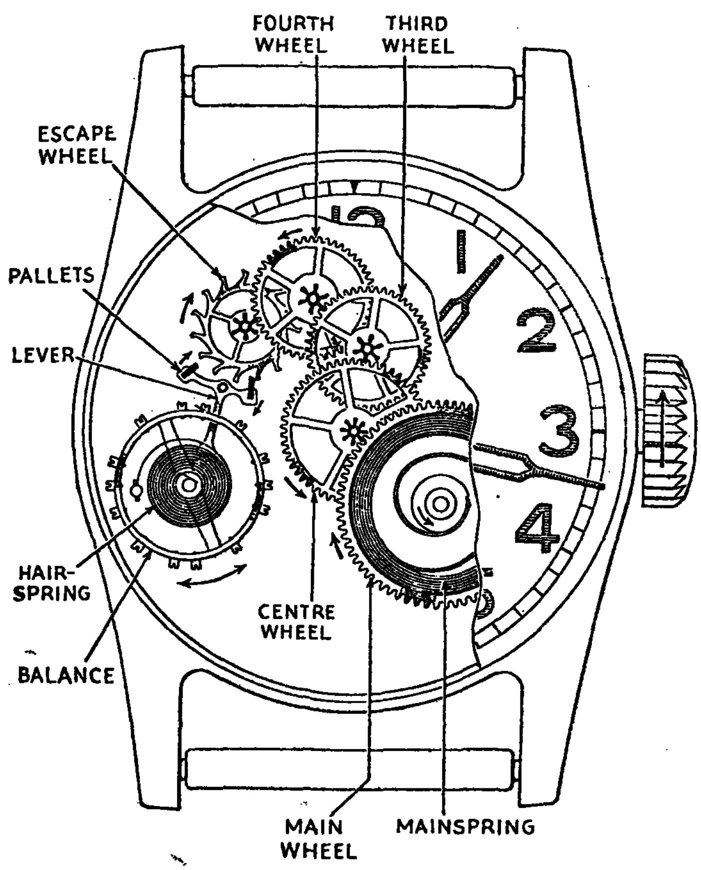
1.3.8 Example Eight

Figure 9 illustrates the situation.

->What happens if the main spring is compressed?

The sticks will move clockwise around the main shaft.

Fig. 9. A Mechanical Clock
This example is the final test for the theory being developed.



2. Perspectives

Here the issues raised in this proposal are examined in the light of AI work on Naive Physics, spatial reasoning, and problem solving.

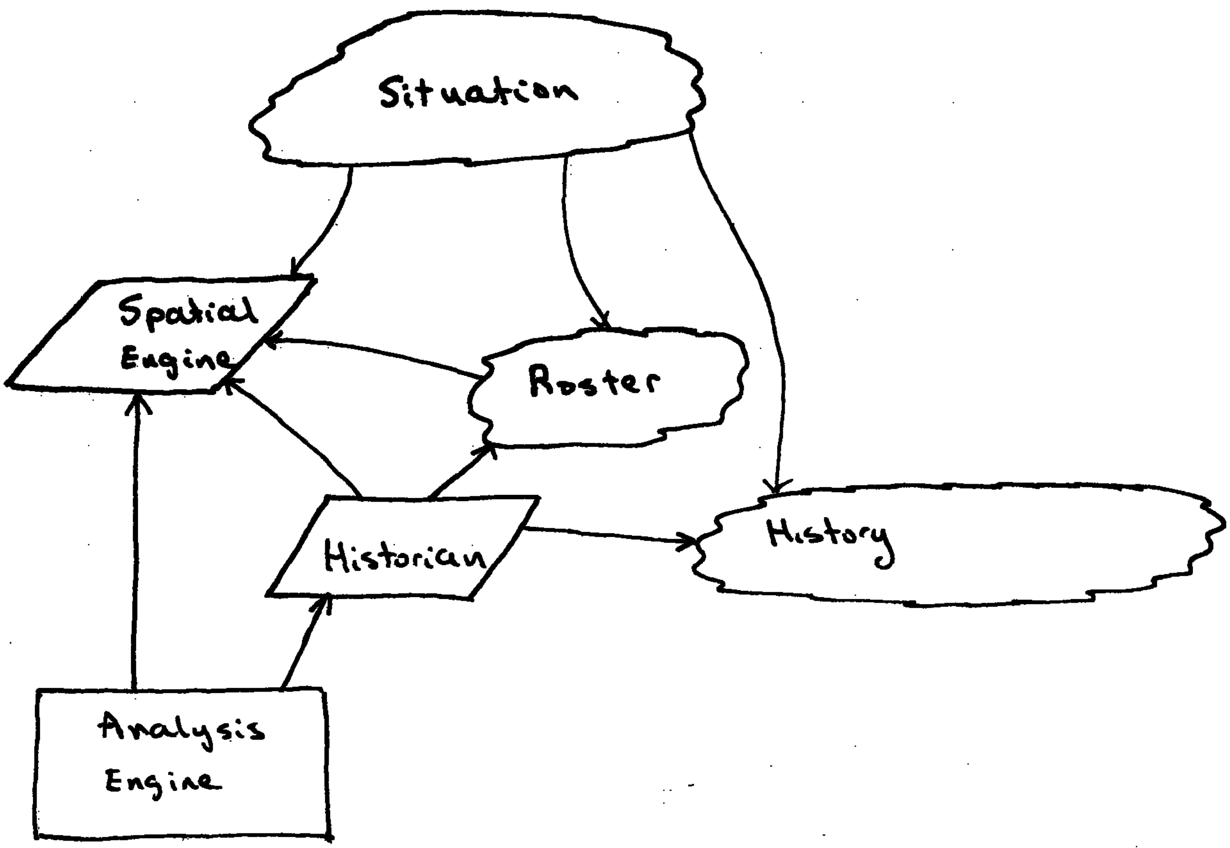
2.1 Naive Physics

The term Naive Physics was coined by Pat Hayes [13] to describe his effort to formalize the common sense thinking we do about the physical world. The manifesto calls for creating a richly detailed axiomatization of what we know about the physical world. He argues that only a very "dense" set of axioms (a large number of connections between the terms of the theory) will be adequate. He also claims that natural module boundaries exist within this set that define "clusters" of related facts. Aside from being important in its own right he argues that such a theory would be a good benchmark for testing inference schemes and representation languages. His plan is to focus on epistemological questions and ignore computational issues.

My past work and the present proposal can be considered as a variation on this theme. The departure concerns computational issues. I believe that common sense reasoning has certain styles associated with it which need to be studied in parallel with the enumeration of the facts they use. Studying how to draw certain inferences can also lead to uncovering facts that should be included but are far from obvious, as the problem of determing the global consequences of assumptions about motion in [8] illustrated.

Experience with FROB and efforts to build a Naive Physics for reasoning about the processes in steam propulsion plants lead me to belive that some very general themes underlie the reasoning processes used in common sense physical reasoning. Figure 10 illustrates a division of labor that seems useful in thinking about such processes, an architecture for Naive Physics machines. The situation description is the statement of the problem that is communciated to the machine. It includes a collection of individuals (written out into the Roster) and the geometry of the situation (which is written out to the Spatial Engine). The Roster keeps track of the indivduals in the situation such as balls, surfaces, or water within a containing space. What these indivduals do is annotated in the Historian. In Naive Physics a history is a description of what an indivdual is doing within explict spatial and temporal bounds[14]. As we will see

Fig. 10. An Architecture for Physical Reasoning
This diagram illustrates a way of breaking up the processes involved in common sense physics.



later on, the tolerance of ambiguity in the Historian determines to a large measure the style of reasoning that can be used. The Spatial Engine is responsible for deductions concerning the geometry, and computes the place vocabulary (see next section) necessary for qualitative spatial reasoning and the Historian. The Analysis Engine uses these representations to solve problems presented to it. When considering how to reason about a class of problems one decides what each of these modules has to do and how they must interact. It is hoped that some of the knowledge needed by these parts and much of the structure of that knowledge is independent of the particular domain within the physics.

The domain implicit in the examples above may be more precisely specified. There are only a small number of qualitatively distinct types of motion. If we ignore fluids, a moving object either (a) mainly touches nothing, (b) has a rotational connection, or (c) has constant contact with a surface. In [8] case (a) was called FLY, case (b) was called SWING, and case (c) is either SLIDE or ROLL[1], depending on the kinds of surfaces involved. Bizarre motions, such as the vertical motion of a ball hung by a rubber band from the ceiling, can probably be shoehorned into this taxonomy (in this case by considering it as an instance of FLY "colliding" with a very odd surface). A result of this thesis would be qualitative theories for the SWING, SLIDE and ROLL cases of motion in three dimensions, thus filling out the taxonomy of qualitative motion theories. Unlike the study of the FLY case, quantitative descriptions of motion will not be built. This should help make the problem manageable.

2.2 Spatial Reasoning

How people deal with shape and space is a deep mystery. We are good enough at spatial reasoning to find our way through a city, tie shoelaces, and bowl. We are not so good that we never crack elbows on unexpected objects, solve jigsaw puzzles instantly, or are able to dispense with layout models in architecture. The common factor which appears in all of the diverse problems we lump into the catagory "spatial reasoning" is the use of a vocabulary of places. A place is a piece of space that has some

1. deKleer studied a simpler version of the SLIDE case in [5] that ignored shape and friction by assuming moving objects were point masses on perfectly smooth surfaces. Aside from introducting the idea of qualitative state and envisioning, interactions with quantitative knowledge and understanding the form of the question were also studied.

interesting property, considered as an individual in the kind of reasoning being performed. Qualitative reasoning about a continuous thing requires quantization of some kind. The quantization cannot be arbitrary;it must define a vocabulary of symbols that is relevant for the kind of problem being solved. One constraint on the quantization are the shapes involved. The domain physics and even computational factors must be take into account as well. Within this point of view, a spatial reasoning problem is fully specified when the relevant place vocabulary and manipulations thereon are defined.

Determining how people represent shape for spatial reasoning purposes is a fascinating open problem. I do not propose to solve it here. As proposed, this work may add a few weak constraints on the representation. There are two major criteria a candidate for human shape representation must satisfy. It must be something that can be computed by the processes involved in perception, and it must be manipulable by the processes involved in spatial reasoning. Much effort has been expended on the first criteria, and some solid reasons exist for using generalized cylinders ([21],[22]. Unfortunately Artificial Intelligence workers have on the whole has ignored the second criteria, preferring to think of the results of vision as useful only for recognition. I think it is critically important to understand the other uses that visual data can be put to. Without pretending to study the exact representation people use for shape, some weak constraints on it still might be obtained. My working hypothesis has been that some significant fraction of the processing involved in spatial reasoning is actually performed by the visual system. The separate nature of the perceptual processes makes it possible to think of the geometric processing as occuring through a data abstraction - whatever the representation is, it is accessed only through certain defined channels. These channels are the questions that may be asked of the shape representation. A shape representation can be evaluated by how easily it answers a particular class of questions. Therefore knowing what questions we need to ask of a shape representation to perform spatial reasoning constrains the underlying representation.

Artificial Intelligence workers have tackled several forms of spatial reasoning. One kind of problem is <u>mental</u> <u>imagery</u>, including [17], [38], and [15]. Kosslyn [17] proposes a mechanism for imagery computations consisting of an array with a "retina". Questions are answered by "scanning" the array with the "retina". The motivation for this model is a set of psychological experiments carried out by Kosslyn. In [8] I argue against this mechanism on computational grounds, and Hinton [15] has developed elegant demonstrations of phenomena that the array model appears incapable of explaining. Waltz and Boggess

[38] use a representation similar to a Metric Diagram as the target semantics for a natural language understanding system, although the questions they ask of it concern only relationships between parts of a static scene.

Another array mechanism was developed by Funt [12] for reasoning about falling shapes. The "retina" in this case performed its processing in parallel, and was "fixated" according to rules that allow it to simulate a situation. As with Kosslyn's array based system, the problems of finding ways to quantize space relevant for the kinds of problems being solved were ignored in favor of a mechanism that only implicitly represents space and shape. The identification of space with cells in the array and of object motion with changes in the cells of that array means the representation of motion is an incremental time simulation. Thus to determine what will happen in a situation will take effort proportional to the time of the motion, with the increment chosen trading off computational effort for accuracy, and the description of motion consists of a set of "snapshots" of the array. It fails to make explicit the kind of motion occuring as well as any other qualitative descriptions. This kind of simulation is quite inadequate for qualitative physical reasoning.

Recognizing the kind of motion from an incremental description (such as might be delivered by perception) has been studied by Badler [1]. The motion is segmented into descriptions that look much like an Action Sequence. These ideas have been extended by O'Rourke[31], who has developed a system that computes a motion description from several images. Neither system uses the descriptions it computes for further reasoning about a situation. Vania [23] proposes to segment motions a person might perform according to the alignment of the axes of the parts of the description of that person's position. The target vocabulary is suited to the intentions of the person being described (WAVE, NOD, CRAWL) rather than motions that have a simple physical meaning. Intentional descriptions and parsing an incremental time description of motion will not be dealt with here.

The problem of getting around in spaces that we cannot see all at once is dealing with "large scale" space. Kuipers [18] modelled a city by a graph whose nodes represent locations and links represent streets. No global coordinate system was used, and little metric information. McDermott [26], on the other hand, explicitly tackles the problem of putting together different coordinate systems with the additional complication of inexact information. McDermott's work can be though of as providing a "fuzzy" representation for placement. While it is more or less reasonable for dealing with large scale space I believe "fuzzification" is a bad idea for touch computations.[1]

A recent representation for the constraints imposed on the placement of an object by the shape of the other objects in its environment is the use of Configuration Space Obstacles [20]. It elegantly solves the classic FINDSPACE problem and closely related problems relevant to robotics (such as FINDPATH) as well.[2] In the framework presented here it may be considered as a way of describing of the place vocabulary for placement problems. Placement, however, is only part of the considerations for the place vocabulary necessary for motion problems (see [8] for the computational and physical constraints imposed by reasoning about motion through free space). While a useful tool for analyzing the constraints shape imposes on motion, it is not clear that algorithms based directly on the Configuration Space formalism will be computationaly tractable (see for example the discussion on Interaction Places below). Configuration space is not a quick solution to all problems involving spatial reasoning.

A major problem in dealing with space are the combinatoric problems inherent in dealing with three dimensions. There is an essential change between one and two dimensions - in one dimension, if one point is between two others, there is no way to get from one end point to another without crossing the point in between. In two dimensions one can "go around" the point in the middle, but cannot get outside a closed figure without going through it. In three dimensions there are even more ways to "go around" in the case of collinear points, and unless the figure consists of closed surfaces a way to leave it may be found. One way to reduce the combinatorics involved in three dimensions would be to reduce a 3D problem to a collection of two dimensional slices. Results obtained in the 2D slices could then be "lifted" back into the three dimensional representation. This is precisely the way Lozano-Perez [20] reduces the computational complexity involved in dealing with the six dimensional configuration space. It is not inconceivable that people perform at least some 3D reasoning in this fashion. Certainly the ability to understand a machine by looking at several drawings of it is suggestive of some ability to perform the "lifting".

---

1. To me "fuzziness" appears to be a phenomenological description of something that is much deeper. Suppose you have a set of processes that purport to give answers to a binary question. If all are in agreement there is no trouble, but what if some of them say "yes" and some of them say "no"? It may be that these cases are precisely those where the phenomena of "fuzziness" presents itself. If so, numerical descriptions of "fuzz" are preventing work on the more deeper problem of identifying the different kinds of knowledge that go into the partial decision processes we have for different problems.

2. So far the details of computing with this representation have been worked out for objects which can be represented by polyhedra or parameterized equations.

This technique makes several assumptions. The most important one is that relevant 2D slices can easily be found. A certain amount of symmetry and simplicity is required in the 3D representation for this to be easy. Finding the relevant slice when shape is represented implicitly by cells turned on in an array, for example, appears to be rather difficult. Evidence from vision [21] argues for 3D representations with this property based on the ability to interpret silhouettes. The work of Lozano-Perez also suggests that computational problems involved need not be intractable. Thus the assumption seems safe. This assumption also seems less reasonable when dealing with the very complicated shapes that spatial modelling systems for robotics require. However, giving precise answers about very complicated shapes does not seem terribly important, since (I venture) people are not very good at that sort of thing. In robotics accurate spatial modelling is required as a replacement for perception. For people (or any other device with a good perceptual system) the real objects can be used to give precise answers to the more complicated questions. The amazing thing about people is that they can give partial answers in complex cases, and pretty good answers in simple cases - there must be something we are doing well, and it would be interesting to find out what that is.

2.3 Problem Solving

Most Artificial Intelligence studies of physics have not been concerned with the kinds of issues studied in Naive Physics. Instead the goal has been to model students solving formal mechanics problems, focussing on such diverse issues as natural language understanding [30] and manipulating equations [4]. While interesting insights have been gained, they have often been at the cost of understanding the deep representations the student has of the physical world(see for example [27],[19]). In FROB the representations involved in physical reasoning were studied at the cost of exploring the control issues involved in problem solving. As it has been argued ([29]), using search seems to distinguish "problem solvers" from other kinds of programs that solve problems. In terms of the Naive Physics architecture above, this means the Analysis Engine should be able to do search if required.

Let us examine deKleer's notion of <u>envisioning</u>([5],[6]). Envisioning is a kind of simulation, based on a qualitative state rather than a numerical state description. Knowledge about motion is encoded into a set of simulation rules that map a qualitative state of motion into a set of qualitative states that may result from it. The mapping is not to a unique qualitative state because of the ambiguity inherent in the qualitative description, for example representing velocity only by a heading (LEFT UP). From an initial state in a situation these rules may be applied repeatedly to produce a description of all the motions possible from the initial state. This graph of qualitative states is called the <u>envisionment</u> for the situation. In more classic AI terms each qualitative state could be regarded as an element in a search space, and envisioning is generating the entire search space for some class of questions. A detailed examination of the limits of envisioning can be found in [9], but this viewpoint should make it clear that there are other options. It is not always necessary or desirable to do that much work. For instance, if no path exists from one place to another an object cannot move between those two places. No intermediate qualitative states of motion need to be generated to get this answer.

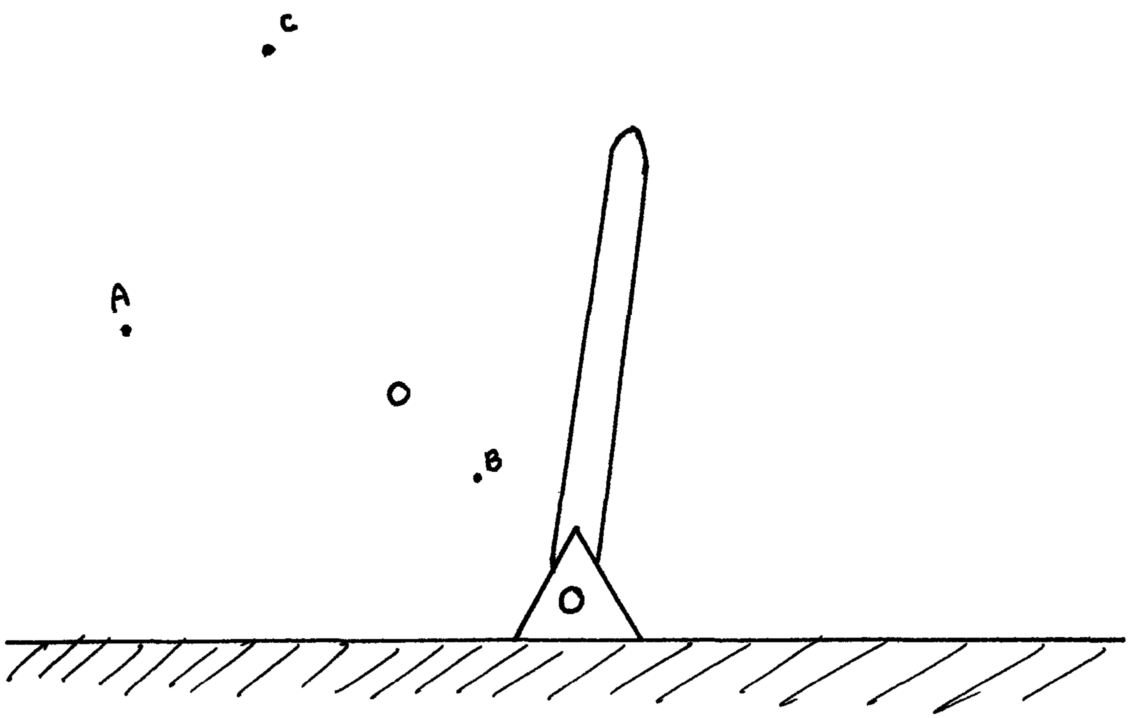
Consider the situation depicted in figure 11. When asked if the tip of the lever can get to point A (or B or C) the answer people give is no. The reasons they give are different in each case. For A the reason is that the pin stops the lever from moving. For B and C however, the reason is that the rigidity of the lever means it cannot bend, stretch, grow, or shrink to reach these points. The simplest envisioning system imaginable would only be able to say that these states are not considered possible. A more subtle one (such as in FROB) would perform the envisioning without the pin and prune the part of the path made impossible by replacing the pin while keeping track of the reason for rejecting that piece of space as a possible location for the lever. To remove the premise rigidity in a similar fashion would complicate the process enourmously.

A simpler way to solve the above problem is to search through the ways the state might happen. If all the theories one can imagine about how a state of affairs might come about can be rejected as being inconsistent with the assumptions of the situation, then that state cannot occur. If one of them is possible, it might occur. In this case, the only way of getting the tip of the lever to B is to move the pivot point, shrinking the lever, or bending it. These three theories are easily eliminated by the assumptions that the pivot was fixed and the lever rigid.

There are two properties of physical knowledge which make this process plausible. The first is that the possible kinds of influences are made explicit and simple, using the notion of a force. The second is that the connections between things in a situation is kept simple and local, as reflected in the bias in physics against action at a distance. These are the same properties that make the qualitative simulation rules work. The qualitative knowledge about motion contained in a program should be cast in forms

Fig. 11. Questions of Excluded States

The pivot is in a fixed position and the lever is rigid. Consider whether or not the tip of the lever can reach A, B and C in turn. The answers people give to these questions are the same that a system relying only on envisionment would give, but the reasons for those answers are quite different.



other than simulation rules so that it can be flexibly used. This will also require explictly characterizing methods according to the kinds of questions they can be used to answer.

At first glance the research proposed here may appear to be similar to the work in electronics, programming, and VLSI done by the Engineering Problem Solving group here at MIT[37][32][36]. Certainly the methods and technology generated in these efforts will prove useful, but the goal here is not to produce a mechanical mechanical engineer. Electronics deals mainly with topological connections. The geometric aspects of VLSI design includes only regular 2D shapes and does not include moving parts.[1] Unlike electronics or programming, our knowledge of mechanical systems explicitly relies on intuition gained by experience with the physical world. Formalizing this intuition is a necessary prerequisite to formalizing the expertise of a mechanical engineer.

One wild ambition implicit in this work is to take a step out from the world of "armchair physics" in which our investigations so often reside. The main reason common sense physics is so interesting is that it is useful in helping us to get around in the world. If we are reasoning in order to deal with the world, we want to be able to make predictions quickly when possible. We want to reason about how some state of affairs could have come about so that it may be duplicated if desirable or avoided if not. We need techniques to see if what we know about a situation is consistent with our physics for we can be mistaken, lied to, or ignorant. Current practice in Artificial Intelligence makes studying these kinds of issues hard. A program is usually told all it will be told about a situation in one initial description and is not allowed to propose actions on the world that would provide more information. They are often designed to provide a quantitative answer in the style of a student solving a physics problem. These restrictions on the design of programs can be valuable simplifications of an already complex task, but if we maintain them too long our efforts may well become distorted.

Aside from an understanding of the basic laws, experience probably provides the judgment of relevant factors required in common sense physical reasoning. It is sensible to assume that the description of objects we begin with when applying our theories about the world are fairly far from the idealizations of the physics. Experience with the world could guide the process of choosing the right physics and mapping from the given objects and relations to the idealzations. Another role for experience

---

1. An exception is a very clever relay which IBM has fabricated out of gold leaf.

is the source of default assumptions. Few people viewing the lever above would volunteer that it would not reach B because the friction in the pivot was so high that the lever could not move at all, yet this could be the case. This aspect of physical reasoning skates close to the deep and turgid waters of learning and does not look simple.

Physical objects can be seen and touched, and while not all of the terms in our theories about them are perceptable (such as density), their effects certainly are. To discover if our theory about a situation is correct we are often goaded into performing actions upon the world. Few people, for example, would claim to fully understand an unfamiliar mechanical gadget just by looking at it; they push and poke its parts to see how it moves, and often modify their theories about it accordingly. A theory of common sense physical reasoning should include theories of what to observe and how to experiment within a situation. It should be able to deduce what the observable consequences of alternate theories concerning a situation are and deduce what sort of manipulations can be made to gain required information.

It is quite possible that just "armchair" reasoning about this domain will soak up all my resources. Even so, developing a deep physical domain such as this one could aid future efforts.

3. Domain Characterization

The world is a very complex place. As usual, to make progress a limited domain must be defined which highlights the important issues and avoids unwanted complexities. An extension of the Blocksworld, called the <u>Toyworld</u>, is defined below. The relevant issues are also catagorized and examined in more detail.

3.1 Toyworld Objects

In the Blocksworld, objects were blocks (and sometimes pyramids and prisms) which existed on an infinite table. The Toyworld includes blocks, but also includes the following primitive solid objects-

1. Cylinders - Circular cross section solid, with finite thickness. Can be used to make sticks and disks.

2. Balls - Spherical solids.

3. Springs - regular helical coiled spring

4. Strings

5. Elastic Bands

6. Holes - allowed in blocks, cylinders and balls. The shape of a hole is in turn that of some block, cylinder, or ball

7. Certain idealizations - such as fixed-position pins and pivots, are allowed when kinematic relations stripped of irrelevant shape are to be expressed.

There are a number of relationships that can hold between objects in the Toyworld. This is illustrated in the examples above. In the blocks world the main relationship was one block being on another. The ways in which Toyworld objects can touch is more complex, since sliding and rolling are to be dealt with. The ability to describe compound objects requires defining ways of combining them, for example by fixing the relative positions between objects ("gluing") and possibly other kinds of connections. Part of the work on the Toyworld lies in developing a useful catalog of relationships.

3.2 Spatial Reasoning

The spatial component of the domain can be thought of in three related parts:

1. How is the shape of an object represented?

2. How is the placement of an object represented?

3. What is the relevant PLACE vocabulary for the kinds of shapes and motions in the domain?

These problems are obviously interrelated. For example, the PLACE vocabulary depends on the shape[1], and both shape and placement are required to determine touch relationships.

I conjecture that people do not accurately compute intersections when thinking about complicated shapes. This is distinct from being able to <u>see</u> intersecting surfaces. If we can perceive a situation, we do not need to have the complicated hardware that seems to be required for accurate computation of three dimensional intersections. The prevalence of physical model construction when such problems have to be addressed (in architecture or mechanical design, for example) is mildly suggestive evidence for this view.

Although people do not seem to be able to provide a definite yes-no answer on whether two complicated shapes will intersect when given their locations and orientations in space, they seem to have the ability to handle some simple cases and to give partial answers in more complex situation. Knowing the length of the teeth and the diameters of two gears, for example, tells us the distance their center shafts must be apart for them to mesh. We also know that if two arbitrarly complex objects, each of which can be enclosed in a ball of a certain fixed radius, have their centers at more than twice that fixed distance then they cannot touch. The property we are looking for in our computations is that of a <u>partial</u> decision proceedure. I think the geometric representation that will deliver this kind of performance is one with quantitative placement information and surface descriptions drawn from a qualitative vocabulary.

---

1. The domain used in FROB provides an illustration. In the Bouncing Ball world balls were represented as point masses, so only surfaces had shape. Introducing curved surfaces to the domain would have only required adding points where the normals were vertical or horizontal to the set of corner points. Giving the balls a finite size would require identifying wells or channels where a ball could not fit, and thus could require a different Space Graph for each ball. Letting the shape be other than round would complicate matters still further.

The value of a quantitative geometric representation was illustrated in [8]. To define the PLACE vocabulary some divisions of free space must be made, and the divisions caused by different objects must be compared to determine how their motions can effect each other. The situations we are modelling are those that we can see; this means "fuzzy" placement as in McDermott's Fuzzy Map are inappropriate.[1] Exact surface descriptions, on the other hand, do not seem appropriate. Consider the quantitative surface descriptions used in computer aided design (such as cubic splines). It does not appear important to know exactly what degree of polynomial the surface is, only that it is, say, smooth and convex. For the qualitative reasoning of interest here, the only properties needed seem to be whether the surface is concave, convex, or planar within some region, whether the surface normal is continuous with the normal of a neighboring region, and possibly what the direction of the normal is at the boundaries between regions.

It might seem that only keeping qualitative curvature descriptions slights the role of surfaces in determining touch relationships. After all, once two objects are placed it is the exact positions of these surfaces that determine whether or not they touch. But this form of the question appears rare for reasoning in this domain. Suppose one object is directly above another, with no third object in between. Since both sets of surfaces are impermeable, they either are touching or are about to touch. The relevant problem is determining what kind of contact is occuring, for this determines how the object will respond to forces placed on it (by sliding or rolling, say) and whether or not it is stable in that attitude. The relevant distinctions seem to be point-point, point-surface, and surface-surface, but the utility of these classes is an empirical question.

3.2.1 2D world

The development of a two dimensional version of the Toyworld would still allow many of the interesting issues of the domain to be studied. The hypothesis about the use of 2D slices in reasoning about 3D situations also requires developing a 2D version of the domain. This is not necessarily simple.

---

1. Note however that the placement of part of a compound object can become inexact if its relation to the whole of the object is specified only in terms of a relationship with a qualitative surface description. I take this as an argument against such local specification rather than an argument for "fuzzy" placement.

The main complexity comes from the requirement of approximating shapes. Imagine a string tied between two points that is longer than the direct distance between the points, so that it hangs down. Few would claim that what we imagine is so detailed that if we were to place a finger at the point in a similar physical situation at the place where in our mental representation we thought there would be string, then there would be string there.[1] But there are places where we are sure string is not, and places where we think string could well be. We have at best a partial decision process for the question "Is there string at this point?". I see no reason to posit an exact description of the string in this case. And if we have the ability to reason with spatial decision procedures which are partial, then that same ability should be useful in dealing with qualitatively simple approximations for other shapes as well.

The basic 2D representation will be a Metric Diagram, but the Metric Diagram implemented for FROB is too simple for the Toyworld. The basic vocabulary must include points, segments, circles, arcs of circles, and polygonal regions. Some kind of approximation hierarchy must be established to represent objects like strings and bands and springs which do not have simple analytic approximations for most cases. The result will be a "geometry box" that combines quantitative placement information and qualitative shape information, as argued for above.

3.2.2 3D world

Assuming the problems of the 2D toyworld are solved and that the hypothesis about the utility of 2D slices is correct, then the main issue of reasoning in 3D situations becomes shape. Since we do not know how people represent shape, we must choose some manipulable representation that we do know about. The information required to reason about more complex shapes must at least include what is required to reason about trivial shapes, so something will be learned even if very gross approximations are used.

Work in vision and robotics has produced several representations for shape. Two popular ones are generalized cylinders [3][16] and polyhedra [20]. At this stage my preference is for generalized

---

1. It may be quite a different matter if we had just seen that particular physical situation. I also suspect the incidence of "hits" in this experiment would be higher if the subject was trained artist.

cylinders. The first reason is that roundness is a requirement for rolling, making polyhedral approximations appear ill-suited. The other reason is that most of the objects in complex Toyworld situations (such as Example Eight) have definite axes, which are important to understand their motion. The relevant slices will likely be a subset of the slices induced by (1) the direction of external applied forces, (2) directions parallel to the axis of some cylinder in the situations, or (3) directions perpendicular to the axis of some cylinder. A potential drawback is that if the class of cylinders chosen is too small (as determined by choices for axes, cross section and scaling functions), the pieces of free space that surround them may not be representable in that class.[1] I doubt this will be onerous for the situations I am dealing with, where quick exact solutions to intersection questions of aribtrarily placed objects are not required.

3.3 Materials

The physical interpretations we make of geometry in our common sense physical reasoning include theories about the stuff objects are made of. Surface smoothness is a theory about what the shape of a surface is at a resolution beyond what is common, and of course is required to estimate friction. Two objects with the same volume but different mass must be made of different "stuff" because the densities are different. These additional details about rigid bodies are important to understand.

The Toyworld includes objects which are not rigid (strings, springs and elastic bands), and the implications for motion of different kinds of non-rigidity must be made explicit. Some time ago Minsky noted that no program then understood that you can pull with a string, but not push with it. As far as I know that is still true.

The integrity of even rigid objects can be violated if enough force is applied. Objects can be bent, stretched, compressed, broken, crushed, or torn. These conditions are at the boundary of where our theory of rigidity lies. Although we typically do not know a detailed account of what happens within them, we are at least aware of how these boundary conditions might be reached and some of the consequences of so doing. A qualitative theory of the effects on motion these changes can cause is an important prerequisite for later mechanical design expertise.[2]

---

1. Personal communication, Tomas Lozano-Perez
2. The exact nature of the changing stress patterns on the Tacoma bridge, for example, were not a part of the common sense understanding of the engineers who built it. Sadly, it was also not a part of their quantitative analysis.

4. Processing

Human reasoning is very flexible, at least by contrast with what we have been able to do with machines. We are fairly efficient, somewhat open to introspection, and are capable of using an accumulation of weak evidence to draw strong conclusions. The struggle to formalize reasoning has gone on for many years and will likely go on for many more. However, I think the inferences required in this project are manageable. This section breaks down the processing that I think needs to occur, including the kinds of questions to be answered, the spatial reasoning problems involved, the "styles" of reasoning required, and some implementation ideas.

4.1 Catalog of Questions

Some of the questions in the examples in the introduction concern only a single situation. These include queries about the value (or possible values) of a property of some object. The single situation questions about motion are variants on:

1. *Response* - If I poke this thing in this direction, what will move and how? (Examples One, Six, and Seven)

2. *Stability* - If I don't poke anything, will anything change? (Example Five)

3. *Summary* - If that thing is moving, what will it eventually do and where will it end up? (Examples Four and Five)

The other kinds of questions require relating the basic situation either to changes of properties in it or to possibly related states. Many of the questions in the examples of the first section (Particularly Examples Two and Three) are of this form. A precise taxonomy of these questions has not yet been developed.

From the examples presented earlier it should be clear that quantitative answers are not sought. What is sought is the ability to come to some kind of conclusion with even weak information, which appears to be an important property of common sense reasoning. I suspect this flexibility comes in part from having a large variety of partial decision procedures; this research will help test that supposition.

4.2 The Spatial Problems

A major portion of the inferences in this domain directly concern a situation geometry and its interpretation as physical objects. Many of the answers forseen are given in geometric terms. Therefore it is appropriate to examine the kinds of reasoning the geometry representation will be used in. The general problem we have is: Given a situation, where and how can the objects in it move? A good problem solving strategy is to divide a complex problem with lots of interconnected pieces into a collection of local problems which can then be assembled into a global solution. The division below attempts to follow this strategy.

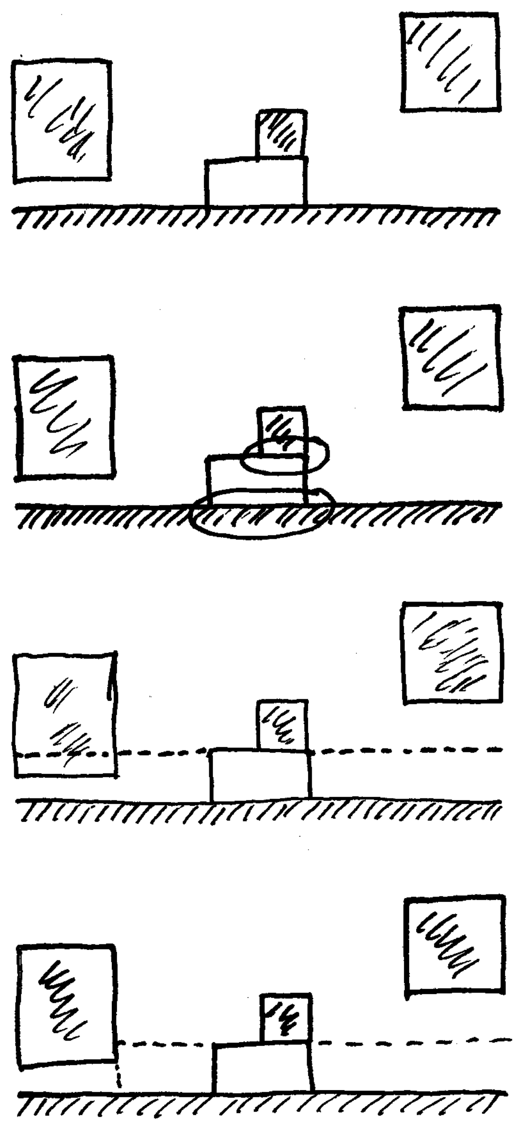
4.2.1 Local Kinematic descriptions

The first problem is to determine, from the shape of an object and what it directly touches, what kinds of motion it can undergo and where it can do so. This will be called the Local Kinematics problem.

What an object touches is determined both by explict connections (such as a string glued to it) and by its placement within the situation. Computing the intersections required to determine the initial touch relationships may be the most expensive part of the geometric computation for a scene. Based on the connections the types of motion can be determined (a rotational connection implies a possibility for swinging, being on top of another object with surface-surface contact makes for sliding, etc). For each type of motion a set of geometric limits based solely on the direct connections should be computed. This set of places can then be pruned by static relationships to other objects, as illustrated in Figure 12. This problem should be tractable because of the constrained nature of the possible motions.[1]

1. The reason for excluding FLY types of motions should be clear at this point. Such computation would require finding the equivalent of a Space Graph for each possible flying object. If these places are exact the description could end up as complicated as a configuration space representation, with additional divisions due to the computational and physical constraints. I suspect that people operate in such situations by using shape only to close off certain places due to size, and consider the object for most of its travel as a point object. The rules for collisions with a surface, of course, become much more complex.

Fig. 12. Local Kinematics
The first drawing shows the basic situation. The second drawing highlights the contact relationships. The third drawing shows the local place for sliding, and the last shows the new places computed by pruning with static constraints. The local kinematic description consists of these places annotated with the type of motion they represent.



4.2.2 Stability

Stability testing requires using the places of the local kinematic description to determine if motion will occur because of the force imposed on the objects by gravity. Much research has been performed on this question for complicated exact shape descriptions because of its relevance to automation. A simpler set of methods for the blocks world was explored by Fahlman [7]. His technique views stability as a problem of finding "chains of support" for each block, with the table providing the "premises" for the support. This idea is both natural and useful. However, he also introduces numerical models for friction and chooses to solve cases of mutual support by iteration. Instead of viewing the problem of stability as finding a consistent assignment of forces or detecting an inconsistency, a classical mathematics view, it could be profitable to view the problem as being one of manipulating dependencies. To wit, what can be locally computed is static stability - "if this set of things doesn't move, this object won't either". Propagation up through these static connections can then determine the stability of other objects. Situations of mutual support would correspond to cycles in the stability relationships. These cycles could either be plunked or analyzed explictly by using the physics to determine an answer.

4.2.3 Determining System Motion

To figure out how some collection of objects moves an account must be given of how the motion of one part causes another part to move. The spatial component of this problem is determining interaction places, that subset of the local kinematic descriptions of two objects that indicates where they might touch. Interaction places can be computed by finding intersections between the local kinematic descriptions for the objects involved. This is much simpler than using configuration space objects. If the local kinematic description were a path along the configuration space of the moving object, a new configuration space object would have to be computed for each pair of moving objects that were to be tested for interaction. The object that was considered fixed in this computation would have to be placed in all of its possible locations and tested for intersections with the configuration space of the object considered moving. For a more detailed description of the placement of objects when they finally touch this much work might be necessary, but even then the search could probably be constrained by the use of interaction places.

In highly constrained situations (i.e., mechanisms) the effects of initial conditions can probably be determined by propagation techniques. In more complicated situations (such as the block and pendulum in Figure 13) a description of possible cases could be generated to guide further analysis. In terribly complicated situations like that of Example Four only the vaguest descriptions of the motions involved can be made. This stage of the computation should be the last where significant geometric analysis occurs. The boundary conditions found in this stage can be used to determine the limit points in quantity spaces which guide further inferences (see below).

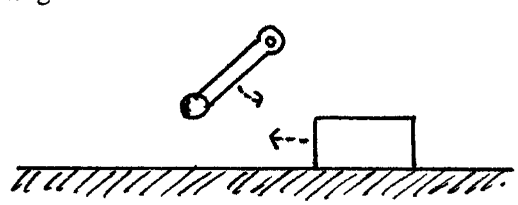
4.3 Knowledge about motion

In classical mechanics the motion of some object is described in terms of its state parameters. A similar notion of state is useful for qualitative reasoning about motion. A qualitative state (or Qstate) abstracts the normal state properties of position and velocity and makes the kind of motion occuring, which is implicit in the choice of equations in classical mechanics, explicit. Qualitative reasoning about motion consists of generating and manipulating these state descriptions.

In NEWTON and FROB most of the qualitative knowledge about motion is encoded in a set of simulation rules. Unlike classical simulation, these rules usually map a Qstate into a set of Qstates due to the abstract nature of the descriptions involved. Simulation rules alone are inadequate. For example, certain questions in the scenarios above could only be answered if there were versions of the rules that ran backwards. Other questions concern properties of collections of states that can probably be answered without explicit generation of that collection. I think the proper way to encode this kind of knowledge is a set of explicit theories that connect two states. Such theories could still be used for simulation but could

---

Fig. 13. A complicated interaction
The kinds of motion the pendulum and block can undergo due to their interaction are quite large.



also be used in simpler ways.

4.4 Styles of Reasoning

Using envisioning corresponds to a particular choice of control structure for qualitative reasoning. It requires a maximum tolerance for ambiguity in the Historian because each state and possible state are elements in the history for that object. Complex manipulations are required to make it reflect the consequences of certain global assumptions about motion [8]. By analogy with problem spaces in classical AI, different problems may be better solved by different search stratagies. The scenarios in the introduction illustrate this. In Examples One and Two the questions involve little or no search. Example Three uses default assumptions to prune the possible Qstates from the initial Qstate until a unique answer is obtained.

Search is not the entire story. In Example One an external influence was excluded by assumption. Several of the answers in Example Two may require search to find, but in the structure of the physical theory (what kinds of things can serve as pendulums) rather than the Qstate space. Example Three concerns relating what will happen to the objects in the situation to a physical parameter which is not directly observable. Not all interesting questions concern relationships between Qstates within a fixed simple physical theory, as noted above. Characterizing the "style" of these other reasoning processes appears important.

To move physical reasoning "out of the armchair" issues involved with observation and experimentation must be addressed. Some experimentation in this area could be performed by annotating the domain knowledge as to what is observable and what kinds of manipulations can be performed. When faced with ambiguity, a program might be able to use what happens in the situation to resolve it. Knowing what sort of physical manipulations can be performed could allow the design of an experiment to determine more about the situation. This kind of problem will provide a real test for the notions of dependency directed reasoning that have been developed in AI.

4.5 Qualitative Process Theory

Many of the properties of physical objects and systems can be thought of as continuous parameters. Understanding physical processes such as moving and boiling requires reasoning about these parameters and the changes physical processes cause in in them. I am developing a theory for performing this kind of reasoning, called Qualitative Process theory. It is designed for determining the limits of physical processes and the relationships induced by processes between various quantities in a system. The current state of the theory is described in [11]. I think it will be an important tool for expressing and using the kind of knowledge this domain requires.

In classical mechanics an important step in solving a problem is to sum up the forces in a situation to see their net result, to resolve the forces involved. The corresponding process in QP terms is deciding which influences will dominate when more than one process affects a quantity. In common sense reasoning it appears that deciding the net effect is done by ignoring some of the influences and ordering the rest. The reasoning in Example Three, for instance, could be summarized as "The breaking point of a stick is higher than any force the spring will produce and we can ignore friction, so the assembly won't move".

The notion of a quantity space is a convinient way to express the limits of rigidity and the boundary conditions of motion. Whether something will bend or break is a function of how much force is applied to it. This threshold becomes a point in a quantity space for static forces, and thus becomes a possible outcome of pushing or pulling the object. Although the basic idea of a quantity space is one dimensional, the boundary conditions of motion may still find a useful expression within it. Assuming a well defined place vocabulary, a particular motion will change from one place to a finite set of other places. It may be useful to place the names of these places in the position quantity space for the object, allowing the indeterminacy of the two dimensional motion to be reflected in the lack of ordering between the boundary points.

Reasoning about causality involves reasoning about functional dependence, and the ways one thing affects another can be thought of as processes. To say that a change in one part causes a change in another is in some sense to say that a change in the property of the second part is a function of the change in some property in the first part. Thus understanding and summarizing the relationship between the parts of a mechanism can be done using the QP theory. Reasoning about other kinds of of functional relationships might become tractable as well - if, say, the pendulum in Example Two breaks when it hits the pin, we can deduce that the faster it is travelling in its arc, the farther left it will fly when it breaks.

4.6 Implementation

The preceeding parts of this section discussed what kinds of processing are required. This section sketches the mechanisms that might perform it. The classic representation language is logic, and first ways to compute with it efficiently are discussed. The localness of constraint propagation means that not all computations can be directly and easily performed in this manner, so interfaces to specialized facilities (which in effect create new "localities") are described. The mechanisms described in this section are evolutionary rather than revolutionary; they may not be adequate for the goals of the project.

4.6.1 Using Logic

The first formal model of thought was logic. Originally developed to classify arguments and capture the semantics of reasoning, it has become a tool of considerable power in the hands of philosophers and mathematicians. I believe that logic is only a subset of the kinds of things involved in reasoning. One cannot ignore issues such as belief revision and the sources of information. However, I would agree with Moore [28] that it is important to understand how to express the kinds of things logic can state well.

There is some dispute about how useful logic is to AI, based largely on experience with theorem provers. I believe the problem with theorem provers comes not from logic but from the particular processing technique chosen to work with it.

Theorem provers mimic the output behaviour of a person doing a proof, down to expressing their knowledge in a syntax suitable for the printed page. The problem with theorem provers is that they generate large numbers of logical assertions even when trying to perform relatively simple proofs, and degrade badly when more information (i.e., more axioms and/or premises) is added. Because of the large number of facts we seem to know about the world, combinatorial explosions do not instill confidence in theorem proving as a model for human reasoning.

Contrast the operation of a theorem prover with a person solving equations. Unless a search or iteration scheme is being used, adding more consistent information makes the answer better, not impossible to find! Part of the problem appears to be that the introduction of new structure is uncontrolled. Richard Brown [2] was aware of this problem and exhibited a rule based system that avoids building structure whenever it can. The Method-Acceptor discipline used in Shrobe's version of AMORD [33] can also be viewed as an attempt to make this distinction. It makes no sense in a theorem prover to speak of "filling in" some facts about an object as being any different from starting on a totally new tack - both consist of making assertions in some kind of data base.[1]

The internal format of a logical statement should be chosen to make reasoning with it easy. There should be some kind of distinction between deductions that are "simple" and those that are "hard", so that the simple ones may be tried first. A possible characterization of the simple deductions are those that could be generated by forward deduction within a network. This is the gist of McAllester's work on propositional reasoning[25]. He turns a set of logical formulae into a constraint network much as [34] turn algebraic models of a circuit into a network, and for the same reasons. Each logical connective becomes an object, connected to cells which represent the propositions. Placing a truth value into a cell results in more cells of the network being filled in as a result of local rules attached to the representations of the connectives. This scheme has the properties we seek. Certain of the inferences will correspond to filling out the network, and others to setting it up and adding parts when required. There is a clear separation between things which are "simple" (pure propositonal reasoning) and those which are "hard" (expanding a description of the situation by adding parts to the network).

Much of the power of logic comes from quantification. McAllester's proposal for encoding universal quantifiers is to view them as specifying additional parts of a network that is "wired in" when the current network has a subgraph that matches the parrticular pattern. Not all universal quantifiers can be viewed in this way (see the axioms in [14] for instance). Some kind of careful control must be exercised over the instantiation process or else the same problem of explosion exemplified by theorem

1. I consider arguments about "cheap memory" absurd in this context. Changing the contents of a memory takes work, so infinite memory would require infinite computational resources - an assumption few are willing to grant in this context.

provers will occur. Controlling this process should probably come under the control of the reasoning program. Existential quantifiers are more problematic. In some situations an extension of a technique used in my version of CONLAN [10] has been useful. In my CONLAN, some cells are considered to hold other pieces of the constraint network as values and for referencing are treated as "indirects" so that constraints can take over much of the burden of connecting themselves. By adding a flag to an indirect that is interpreted to assert whether or not it is considered possible for the value to exist many deductions that in standard notation would use existential quantifiers can be made. What is lacking is the ability to instantiate an arbitrary, untyped individual and the ability to determine what type this arbitrary thing is from the information collected about it by virtue of its descriptions. A better solution needs to be found, possibly along the lines of the Steele's XPRT system[35].

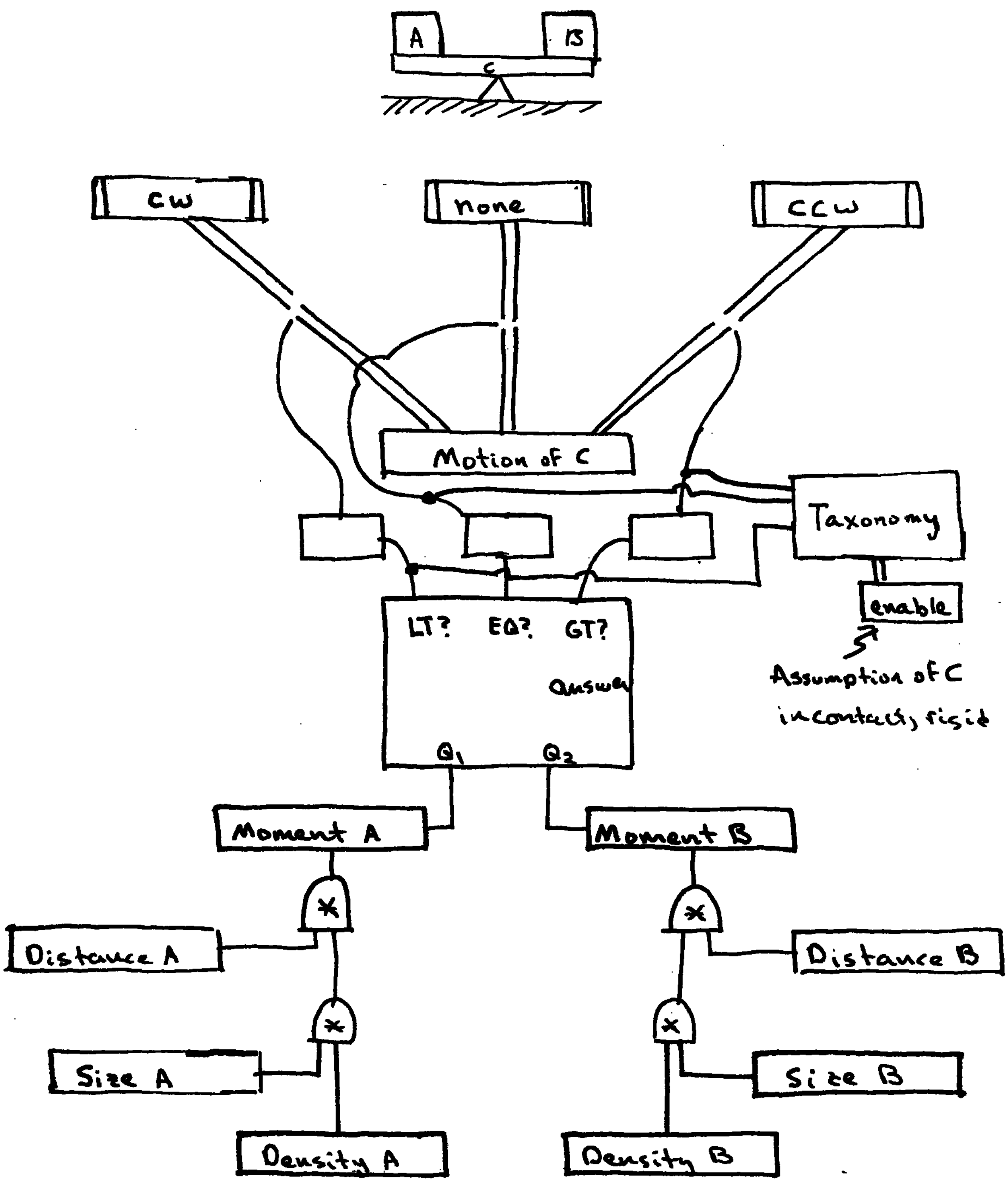
4.6.2 Specialized Inference Engines

While the basic semantics of first order logic is an important part of the power this project requires, simplifications will result from not restricting ourselves to it. Some patterns of statements occur so often that it is best to think of them as a new operator. In Hayes' Naive Physics for liquids[14], for example, the Taxonomy statement is introduced. A taxonomy statement is true if and only if exactly one of its clauses is true. Although its semantics may be expressed by some combination of other logical statements, is so useful that it merits seperate status. In other words, it appears to reflect "module boundaries" that are useful in our own thinking. It is easy to write a constraint prototype to enforce these semantics. These statements are useful for expressing a fixed set of alternatives(see Figure 14). There are probably other statements that capture certain patterns of human reasoning, and exploring the Toyworld might be a good way to find them.

Some kinds of reasoning, such as geometric computations or inequality questions, are better performed by special purpose inference engines. These must interact smoothly with the logical portion of the system. This can be done by using some of the elements of a constraint network to act as interfaces to specialized modules that can answer particular kinds of questions. In FROB, for example, some constraints interfaced to the Metric Diagram that was used for most geometric reasoning. A Metric Diagram will also be required here, although the interface with the constraint system must be more

Fig. 14. Network describing a See-Saw

To be flexible in using partial information, a network must contain reasonable detail and use "reversable" components. This network captures the relationships between the physical parameters of the See-Saw and the motions that can result. The assumptions which lie outside the network are (a) that C remains in contact with D (i.e., is not lifted), (b) that A and B remain in contact with C, and (c) that C and D are adequately rigid.



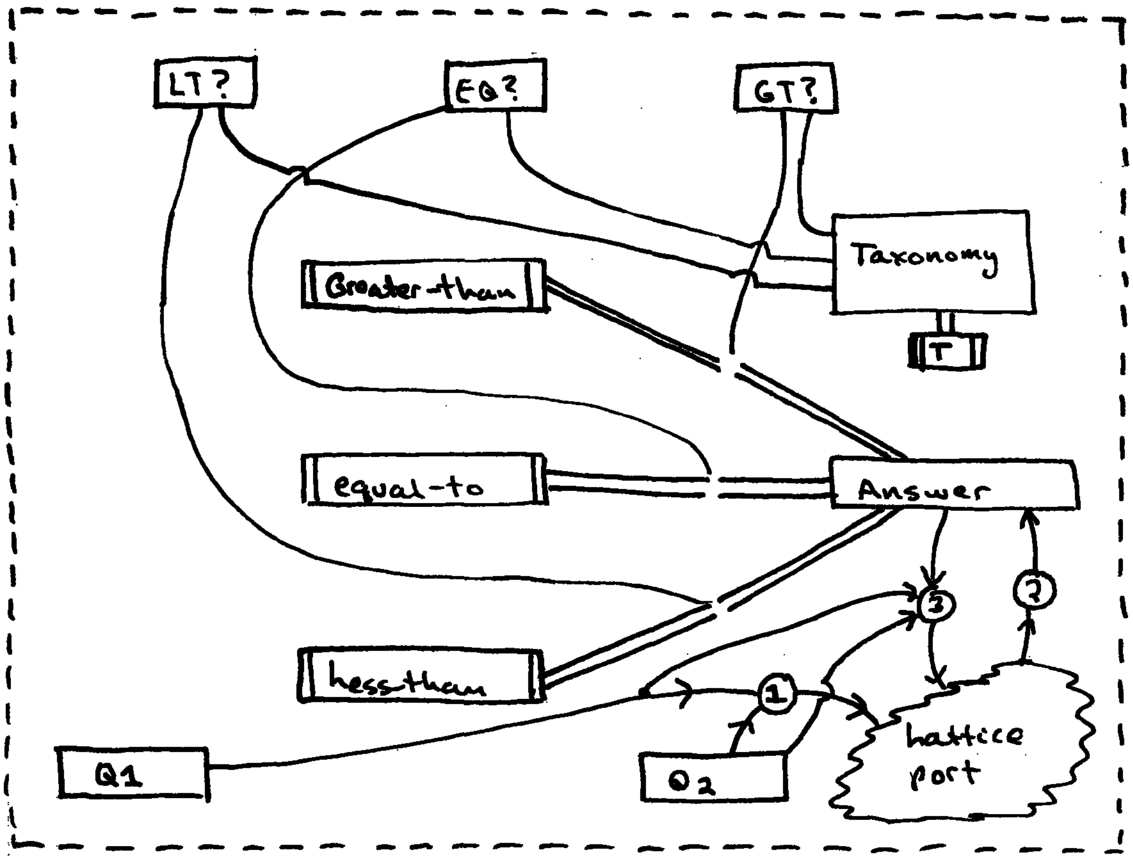
flexible.

A module that is essential to the efficient implementation of the Qualitative Process theory is an inequality expert. Inequalities are often considered tricky to reason about because of the transitivity of their axioms. A program which maintains a lattice model corresponding to the inequalities which have been asserted in the description can answer questions about them quite easily. Figure 15 contains a schematic of a constraint that has been used to run such a module. This constraint is used both to write discovered inequality relationships into the lattice and to access those already known. Dependency information is maintained to keep the information in the lattice and the constraint network consistent.

Summing the influences on a quantity will probably require a special module that describes the sum of a set. The possible members of this set are determined by the "connectivity" of the situation. Some of the members may be ruled out by assumptions (such as when ignoring friction). The others must be put together somehow, and since the result is the IQ value as well as the rate of change for the quantity even partial information should be computed if possible. Using the inequality expertise could allow the determination of the sign of the result in some cases where otherwise nothing could be determined. If for example each negative element can be matched with a larger positive element, and there are at least as many positive elements as negative elements, then the sum is positive. Stated formally, this problem is equivalent to finding a subgraph of a given bipartite graph (where the nodes are quantities in the sum and the links are inequalities).

Fig. 15. Computing with inequalities

Often in qualitative reasoning only relative magnitudes of certain quantities are known. A special module can be used to decide questions about partial orders with reasonable efficency. Q1 and Q2 are the symbolic quantities, and the answer terminal describes the relationship that holds between them. The other three terminals are connected to the answer as a taxonomic choice (i.e., exactly one of them must be the answer), and are used in "decoding" the answer and in accumulating partial information.



Rule 1 places a request in the lattice for order information

Rule 2 returns an answer from the lattice (with dependency information) if it ever arives

Rule 3 places the answer into the lattice if it is found by other methods (such as exclusion)

5. Plan of Attack

It is traditional for thesis proposals to include some kind of research plan, complete with schedules, time estimates, and milestones. In keeping with the current focus on qualitative reasoning, no metric information about times will be provided. The work divides nicely into three themes which split off from where I am now, interacting in several places. These themes are elucidated below and their interdependence noted. The ambitious scheme is to consider the thesis done when these three themes unite in computing a description of a mechanical clock. In other words, given a description of a clock and told that the spring is wound the program should determine that the hands will move, and not by a method detailed enough to allow it to deduce that it will lose only a minute a month!

5.1 2D Toyworld Development

Defining a two dimensional version of the Toyworld and building a program to reason about it is interesting for two reasons. First, if my conjecture about 3D reasoning is correct it will be a major portion of the full theory. But even if the conjecture is wrong, there are still interesting kinds of reasoning to study in it, as the scenarios illustrate. The first step is to develop an adequate geometry representation, along the lines of the Metric Diagram but including approximations. The spatial part of the theory will evolve from there, and implementation will proceed along with the theorizing. The general sequence that seems appropriate is first local kinematics, statics, and finally dynamics. In parallel with the spatial development appropriate theories about materials can be formulated.

5.2 3D Reasoning

In parallel with the development of the 2D Toyworld, alternatives for representing the shapes in a 3D version need to be examined. Once the 2D kinematics has been developed an effort to develop a 3D geometry box with projection and lift capabilities will be started. Three dimensional theories of statics and dynamics should evolve just after their corresponding two dimensional version.

5.3 Qualitative Reasoning

A deeper analysis of what is involved in answering the questions presented in the scenarios needs to be performed. A taxonomy of relevant questions about the domain will be constructed for use as a precise check of the adequacy of whatever theories get embodied in programs. The muddied area of the different "styles" of reasoning needs to be thought about further, and related to the types of questions to be answered. This line of development corresponds to deciding what is in the Analysis Engine mentioned above.

Two important issues in reasoning about motion that appear in this domain are detecting oscillation and determining the impact of friction. Oscillation detection in NEWTON and FROB depended on qualitative simulation to produce a cycle of Qstates in the envisionment. It should be possible to use considerations of symmetry to infer oscillations in some sitations. Unlike the Bouncing Ball world, friction in the Toyworld has a constant impact on motion. Usually it can be ignored, but we also have a sense of how changes in friction will change a situation. Studying these issues require using the results of the spatial reasoning components as they are built. In terms of the Naive Physics Architecture, this part corresponds with specialized development of the Historian and Analysis Engine.

Finally the proper set of annotations denoting observable and manipulable quantities must be found, and investigations into experiment planning made.

5.4 Time Required

I think this research will take two years.

References

1. Badler, Norman I. "Temporal Scene Analysis: Conceptual Descriptions of Object Movements" Technical Report No. 80, Moore School of Electrical Engineering, University of Pennslyvania, Philadelphia, February, 1975
2. Brown, Richard "Coherent Behavior from Incoherent Knowledge Sources in the Automatic Synthesis of Numerical Computer Programs" TR-610, MIT AI Lab, January 1981
3. Brooks, Rodney and Binford, Thomas "Interpretive Vision and Restriction Graphs" Proceedings of the First Annual National Conference on Artificial Intelligence, AAAI, August 1980
4. Bundy, A. et. al. "MECHO:Year One" Research Report No. 22, Department of Artificial Intelligence, Edinburgh, 1976
5. deKleer, Johan "Qualitative and Quantitative Knowledge in Classical Mechanics" Technical Report 352, MIT AI Lab, Cambridge, Massachusetts, 1975
6. deKleer, Johan "Causal and Teleological Reasoning in Circuit Recognition" MIT AI-TR-529, Cambridge, Massachusetts, September 1979
7. Fahlman, Scott "A Planning System for Robot Construction Tasks" TR-283 MIT AI Lab, May 1973
8. Forbus, K. "A Study of Qualitative and Geometric Knowledge in Reasoning about Motion" TR-615, MIT AI Lab, Cambridge, Massachusetts, February, 1981
9. Forbus, K. "Qualitative Reasoning about Space and Motion" in <u>Mental Models</u>, D. Gentner and A. Stevens, editors. LEA, inc. 1981
10. Forbus, K. "A Conlan Primer" BBN Technical Report No. 4491
11. Forbus, K. "Qualitative Reasoning about Physical Processes" Submitted to IJCAI, 1981
12. Funt, Brian. V. "WHISPER:A Computer Implementation Using Analogues in Reasoning" PH.D. Thesis, University of British Columbia, 1976
13. Hayes, Patrick J. "The Naive Physics Manifesto" in <u>Expert Systems in the Micro-Electronic Age</u>, edited by D. Michie, Edinburgh University press, May 1979
14. [Hayes, b] Hayes, Patrick J. "Naive Physics 1 - Ontology for Liquids" Memo, Centre pour les etudes Semantiques et Cognitives, Geneva, 1979
15. Hinton, G. "Some Demonstrations of the Effects of Structural Descriptions in Mental Imagery" <u>Cognitive Science</u>, Vol. 3, No. 3, July-September 1979
16. Hollerbach, John "Hierarchical Shape Description of Objects by Selection and Modification of Prototypes" TR-346 MIT AI Lab, November 1975
17. Kosslyn and Schwartz, "A Simulation of Visual Imagery" <u>Cognitive Science</u>, Vol. 1, No. 3, July 1977
18. Kuipers, Ben "Representing Knowledge of Large-Scale Space" TR-418 MIT AI Lab, July 1977
19. J. Larkin, J. McDermott, D.P. Simon, and H.A. Simon "Models of Competence in Solving Physics Problems" Cognitive Science, Volume 4 Number 4, October-December 1980
20. Lozano-Perez, Tomas "Spatial Planning with Polyhedral Models" MIT AI Lab Technical Report, September 1980

21. Marr, David "Analysis of Occluding Contour" Proc. R. Soc. London, Biology. Volume 197, p441-475, 1977
22. D. Marr and H.K. Nishihara "Representation and Recognition of the Spatial Organization of Three Dimensional Shapes" Proc. R. Soc. Lond. Biology, 200, 269-294, 1978
23. D. Marr and L. Vania, "Representation and Recognition of the Movements of Shapes" Memo No. 597, MIT AI Lab, October, 1980
24. Mason, M. T. "Compliance and Force Control for Computer Controlled Manipulators", TR-515, MIT AI Lab, April 1979
25. McAllester, David. "An Outlook on Truth Maintenance" MIT AI Lab Memo 551, August, 1980
26. McDermott, Drew "Spatial Inferences with Ground, Metric Formulas on Simple Objects" Yale University Computer Science Research Report 173
27. McDermott, J. and Larkin, J. "Re-representing Textbook Physics Problems" in Proc. of the 2nd National Conference of the Canadian Society for Computational Studies of Intelligence, Toronto 1978
28. Moore, Robert "Reasoning from Incomplete Knowledge in a Procedural Deduction System" TR-347 MIT AI Lab, December 1975
29. A. Newell and H. Simon "Human Problem Solving" Prentice-Hall, Inc. 1972
30. Novak, G. "Computer Understanding of Physics Problems Stated in Natural Language" Technical Report NL-30, Computer Science Department, The University of Texas at Austin, 1976
31. O'Rourke, J. "Image analysis of Human Motion" PhD Dissertation, Department of Computer Science, University of Pennsylvania, 1980
32. C. Rich, H. Shrobe, and R. Waters "Computer Aided Evolutionary Design for Software Engineering" Memo No. 506, MIT AI Lab, January 1979
33. Shrobe, Howard "Dependency Directed Reasoning for Complex Program Understanding" TR 503 MIT AI Lab, April 1979
34. Steele, G. and Sussman, G. "Constraints" Memo No. 502, MIT AI Lab, Cambridge, Massachusetts, November 1978
35. Steels, Luc "Reasoning Modeled as a Society of Communicating Experts" TR-542 MIT AI Lab, June 1979
36. G.J. Sussman, J. Holloway, and T.F. Knight, Jr. "Computer Aided Evolutionary Design for Digital Integrated Systems" Memo No. 526, May 1979
37. Sussman, Gerald J. "Electrical Design: A Problem for Artificial Intelligence Research" Memo No. 425, MIT AI Lab June 1977
38. Waltz, D. and Boggess, L. "Visual Analog Representations for Natural Language Understanding" in Proc. IJCAI-79, Tokyo, Japan, August 1979