Connectivity and Time in Qualitative Simulation

M. DiManzo[1], D. Tezza and E. Trucco

DIST – Univ.of Genova, via all'Opera Pia 11a, 16145 Genova, Italy

[1]Inst.of Comp.Sci.,University of Ancona,via Brecce Bianche,60131 Ancona, Italy

Abstract

In this paper we present a qualitative model for one–dimensional flexible objects taking into account various classes of problems: dynamical, kinematic and temporal. A first implementation of the model, in which only dynamics had been considered, was able to perform prediction, activity determination and skeptical analysis in a simple world composed of strings moving under the action of forces. The research described here has aimed at improving the dynamical model within the framework of Qualitative Process Theory and adding new features. We will be introducing the ontological elements for the domain considered, a logical framework for reasoning about connectivity relationships among the above–mentioned elements, and a mechanism for performing inferences in which time plays a central role. Some examples will illustrate the various aspects of the theory.

1. Introduction.

Space and time are two basic dimensions of reasoning about physical phenomena. In qualitative modelling, the dynamic properties of systems, which can be described, for instance, by means of the Qualitative Process Theory (QPT) [for84] or other related formalisms [de85, kui84], must be linked to their spatial structure (e. g. shapes and connectivity relations), in order to obtain meaningful responses from a qualitative simulation. Spatial constraints appear in QPT as logical prerequisites to the existence of objects and processes; they are assumed to be true, but their possible evolution is not accounted for. Several authors addressed the problem of defining a suitable qualitative representation of space [for83, kui, sho85]. A basic contribution is due to Forbus et al. [for87], who postulates that a purely qualitative description is too poor, and it can be conceived only as an abstraction of un underlying metrics. However, a lot of reasoning can be carried out on this abstract representation, if the metric representation supplies the symbolic description with a consistent semantics. In this paper we suggest the use of a temporal logic formalism for analyzing the temporal evolution of connectivity prerequisites, thus avoiding the explicit definition of a complete diagram of kinematic states.

As regards time, a number of logics have been proposed for dealing with time varying properties [all83, mcd82, sho87]. However, in almost all qualitative models time seems to play an implicit role in the definition of other quantities, rather than being the direct object of reasoning. In this paper we propose an extension of the QPT which allows the evaluation of the evolution of overall properties of episodes, as, for instance, their duration.

The problem faced in this paper have been raised by the simulation of the String World [di87], where a meaningful question is, for example, how to control the period of a pendulum by varying its length. A brief description of a qualitative model of strings is given in section 2, limited to the inextensible string case; a deeper analysis can be found in [di88]. Section 3 is concerned with the prerequisite analysis, and section 4 with temporal reasoning.

2. An ontology for the domain.

Let's start with the specification of the objects which comprise our domain, defined by means of the quantities that characterize their behavior.

Strings

A string refers to whatever one−dimensional flexible object (cables, threads, strings, ecc.). It is simply defined as:

(FOREACH str ∈ *string_segment*
Has_quantity(str,breaking_tension)
∧ Has_quantity(str,length)
∧ Has_quantity(str,left_end) ∧ Position(left_end(str))
∧ Has_quantity(str,right_end) ∧ Position(right_end(str))
∧ Has_quantity(str,left_end_velocity)
∧ Velocity(left_end_velocity(str))
∧ Has_quantity(str,right_end_velocity)
∧ Velocity(right_end_velocity(str))
∧ Has_quantity(str,internal_force) ∧ Force(internal_force(str))
∧ Has_quantity(str,rest_length)
∧ (Unstretchable(t) ⇒ A[length(t)] = A[rest_length(t)]))

Forces, positions and velocities are defined as having a value and a direction.

Supports

A *support* indicates anything which can apply forces to a string, e.g. in order to produce or stop motion, like a robot gripper, or simply to sustain it while it slides, like a pulley. It is through supports that a physical string is discontinued into subsegments with different dynamic properties. These subsegments can be regarded as individual views. In our world, we define two types of supports: *slide_supports* and *block_supports*. Only slide_support are considered in the following.

This class contains all supports which let a string slide with or without friction. Most importantly, slide_supports require the physical continuity of the string over the two subsegments (*continuity*) and impose the same velocity to the adjacent endpoints of the two segments. They work as motion and force propagators between two string subsegments. For simplicity, a slide_support is always supposed to be stationary. Notice incidentally that, according to the model below, the string might slide on the support forming whichever angle and should not necessarily lie in the vertical plane.

```
(FOREACH s ∈ slide_support
        Has_quantity(s, location) ∧ Position(location(s))
        ∧ Has_quantity(s, force) ∧ Force(force(s))
        ∧ Has_quantity(s, static_friction)
        ∧ Force(static_friction(s))
        ∧ Has_quantity(s, dynamic_friction)
        ∧ Force(dynamic_friction(s))
        ∧ (FOREACH t1, t2 ∈ string_segment
            (Continuity(t1,t2) ∧ A[right_end(t1)] = A[location(s)]
            ∧ A[left_end(t2)] = A[location(s)]
            ∧ Contact(t1,s) ∧ Contact(t2,s))
            ⇒ A[right_end_velocity(t1)] = A[left_end_velocity(t2)])))
```

The relation *Continuity* states that two segments are parts of the *same* physical thread, which is in *Contact* (i.e. exchanges a force) with a support. The remaining conditions specify that the contact involves the right end of a segment and the left end of the other. As a consequence, the two end point velocities must be equal.

String segments

String configurations can change in time, e.g. if a catenary is gently let down, when it reaches the ground (a support) we can consider it gone. Such states are modeled with individual views. For example, two basic configurations are the string suspended by its endpoints (*catenary*) and the *taut* string; under the hypothesis of inextensible strings, the corresponding IV's suffice for the description of various scenarios. For the sake of brevity, let us describe only the catenary IV.

```
Individual_view Catenary(t, s1, s2)
Individuals:
        t a string_segment;
        s1, s2 a slide_support or block_support;
Preconditions:
        Contact(t, s1); Contact(t, s2);
        (FOREACH s ∈ slide_support ∪ block_support,
        In_between(s, s1, s2) ⇒ ¬Contact(t,s));
Quantity_conditions:
        A[left_end(t)] = A[locations(s1)];
        A[right_end(t)] = A[locations(s2)];
        A[length(t)] ≧ A[distance(s1, s2)];
Relations:
        A[length(t)] = A[rest_length(t)];
        Let curvature be a quantity
        curvature ∝q+ length(t);
        curvature ∝q- distance(s1, s2);
        Let hmin be a quantity
        hmin ∝q- length(t);
        hmin ∝q+ distance(s1, s2);
```

Notice that *contact(a,b)* indicates not only that a and b are in contact but also that they *exchange a force*. Since connections can be decided within a *qualitative kinematics* (what has been called the *connectivity hypothesis* [for87]), *contact(a,b)* is a precondition and not a quantity condition. Section 2 below deals with this kind of problems.

2.1. Describing the behavior of inextensible strings.

The evolution of a given scenario in the domain considered is described by processes, which will be active in particular situations and will affect the individuals involved. Most processes are concerned with the behavior of the string segments (e. g. a catenary becoming a taut string); the dynamic behavior of blocking support is simply described by two processes *Support_motion* and *Support_acceleration* directly derived from the corresponding processes of motion and acceleration defined by Forbus in [for84].

As a first example of process let's consider the sliding of a catenary. There are neither preconditions nor relations in that they are the same as those holding for the individual view *catenary*.

```
Process Catenary_slide(t,s1,s2)
Individuals:
        t a string_segment;
        s1, s2 a slide_support or block_support;
Quantity_conditions:
        Catenary(t,s1,s2);
        (Am[left_end_velocity(t) > ZERO ∧ A[velocity(s1)] = ZERO)
        ∨(Am[right_end_velocity(t) > ZERO ∧ A[velocity(s2)] = ZERO);
Influences:
        I−(length(t), A[right_end_velocity(t)]);
        I+(length(t), A[left_end_velocity(t)]);
```

Notice that this process describes a *shape change*, though dynamic conditions are considered only in the endpoints.

Another process involving string segments is the sliding of a taut string. Actually two processes are needed for describing this behavior because a taut string must be pulled in different directions at the two endpoints to keep it taut. ("you can pull with a string but not push with it"). Only *sliding_taut_left* is given; its companion follows obviously.

```
Process Sliding_taut_left(t,s1,s2)
Individuals:
        t a string_segment;
        s1, s2 a slide_support or block_support;
Preconditions:
        Slide(s2);
        (∃ t1 ∈ string_segment
        A[left_end(t1)] = A[location(s2)]
        ∧ Continuity(t, t1)
        ∧ Unconstrained(t1));
Quantity_conditions:
        Taut_string(t,s1,s2);
        A[left_end_velocity(t)] < ZERO;
Relations:
        A[right_end_velocity(t)] = A[left_end_velocity(t)];
        Let taut_velocity be a quantity
        A[taut_velocity] = A[left_end_velocity(t)];
Influences:
        I−(length(t), A[right_end_velocity(t)];
        I+(length(t), A[left_end_velocity(t)];
```

The above mentioned IV's and Processes allow us to represent many scenarios in our domain, along with their qualitative behavior (provided all the strings involved are inextensible). For example, we can consider a string blocked by two supports at its end points, in contact with some slide support located in between, and pulled at one end by the blocking support. The envisionment for this scenario will consist of a certain number of IV's catenary (depending on the slide_support number) progressively turning into a single taut string. Since the string is supposed to be inextensible, the support motion will come to an end with a sudden stop resembling very much an inelastic collision, which is easily modeled through an encapsulated history *run_stop* (omitted for the sake of conciseness), which imposes that the endpoint velocities become ZERO. This will disactivate the slide process. Moreover, the *net* forces on the two supports also become ZERO, which in turn disactivates possible acceleration processes, thus leading to a stationary situation. Notice that, if all slide support have the same height, the Contact relationships between the string and the supports themselves don't change; in this case the only events predicted are those of the various catenaries becoming taut strings. However, if a middle support is lower than the others, another important event takes place because of the string sliding, that is the detachment of the string from the lowest slide support; this event can't be envisioned by QPT by itself, in that it involves the failure of a Contact precondition, but it plays a central role in the envisionment for the given scenario (since it leads to a new view structure), and thus it must be correctly inserted in the history. The logic for preconditions and the temporal framework described in the next two sections aim at facing this class of problems.

3. Reasoning about Prerequisites.

It is apparent from the previous dynamic model of the string world that connectivity relationships are essential in our representation: typical preconditions of individual views and processes describe various kinds of Contact relations between string segments and supports. It is also conceivable that the system dynamics may affect such relations, so we must provide the model with some reasoning capability about preconditions.

A possible solution of the problem is attainable by means of the Qualitative Kinematics (QK) framework proposed by Forbus et al. in [for87], in which the notion of kinematic state is introduced based on connectivity relationships. However, although it seems to be able to capture the central role connectivity plays in our model, in our case that theory is not completely satisfactory. The main reason lies in the fact that Forbus' methodology requires finding out all possible connectivity relationships for the domain considered and then consistently combining them in all possible ways to form all consistent global kinematic states.

In our case, in finding out all possible connectivity relationships we must take into account that, a priori, each string might be in contact with each support (albeit not simultaneously). Hence the number of global kinematic states resulting from consistent combinations of connectivities, although reduced by applying constraints based on the maximum string lengths, is often very large, and leads to serious problems in building the QK state diagrams even for simple scenarios in our domain. Moreover the notion of global kinematic state is very sensitive to any local change in the structure of the model. On the other hand, local connections are very simple (a contact exists or not), so a decomposition of the model is very tempting. However, giving up the notion of global kinematic state, we cannot longer rely on the state diagram for constraining allowable state transitions. So, an alternate approach for linking dynamic and kinematic information must be found.

Temporal logic can be used for this purpose. The major advantage of this method is that it doesn't require to compute in advance a (in our case) large number of kinematic states along with the possible transitions between them; only a relatively small number of logical rules will suffice. Given that set of rules, we can match the current view and process structure against it, in order to find out whether the current active processes can lead to some changes in connectivity (Notice that in doing so, we again rely on the sole mechanism assumption [for84], which states that processes are the only entities that can cause directly or indirectly changes in a given situation).

The common aspect of our solution and the kinematic theory proposed by Forbus is the Poverty Conjecture [for87], according to which no purely qualitative general purpose kinematics exists. In other words, in reasoning about kinematic problems, such as spatial relationships between moving objects, purely qualitative information doesn't suffice, but quantitative information is needed as well. In Forbus' work this fact is reflected in the Metric Diagram / Place Vocabulary (MD/PV) model, which combines a qualitative representation provided by the PV with quantitative knowledge provided by the MD. In our case we have a sort of PV represented in a logic form through the above mentioned rules; the MD is attained by means of a global system of coordinates, in which a given scenario is represented. The quantitative (numerical) information supplied by this MD is used to check the actual occurrence of a connectivity change as specified by a certain rule. Finally it is worth noticing that a MD, with its quantitative representation, is needed for our domain (i. e. flexible objects) to simulate the human ability in reasoning about spatial relationship problems by means of the visual system, since that kind of problems is ubiquitous in the domain.

3.1. Rules for Changing Connectivity Relations.

In this section we present some of the rules governing the changes in connectivity. Being aimed at modelling connectivity relationships, the rules must be able to describe both the occurrence of a new contact and the end of a previous contact between a string segment and a slide support (blocking support are not taken into account yet, in that they involve an active action by the support, which can't be predicted within our sole model). The end of a contact relationship can be modeled, for example, by the following two rules (the notation for time is derived from that proposed by Shoham in [sho87]):

```
(FOREACH str ∈ string_segment, FOREACH s ∈ slide_support,
 FOREACH t1, t2 ∈ temporal_term,
   (TRUE(t1, t2, Contact(str,s))
    ∧ (∃ str1 ∈ string_segment, ∃ s1 ∈ slide_support ∪ block_support
       ((TRUE(t1, t2, Taut_string(str1,s,s1))
          ∧ TRUE(t1, t2, Taut_slide_right(str1,s,s1))
          ∧ TRUE(t1, t2, Part_of(str,str1))
          ∧ ¬(∃ str2 ∈ string_segment,
                TRUE(t1, t2, (A[right_end(str2)] = A[location(s)]))
                   ∧ TRUE(t1, t2, Contact(str2,s))))
      ∨ ((TRUE(t1, t2, Taut_string(str1,s1,s))
             ∧ TRUE(t1, t2, Taut_slide_left(str1,s1,s))
             ∧ TRUE(t1, t2, Part_of(str,str1))
             ∧ ¬(∃ str3 ∈ string_segment,
                   TRUE(t1, t2, (A[left_end(str3)] = A[location(s))
                      ∧ TRUE(t1, t2, Contact(str3,s))))
   ⇒ (∃ t3 ∈ temporal_term, t3 > t2,
      TRUE(t2, t3, ¬Contact(str,s))))
```

```
(FOREACH str1 ∈ string_segment, FOREACH str2 ∈ string_segment,
 FOREACH s ∈ slide_support, FOREACH t1, t2 ∈ temporal_term,
  (TRUE(t1, t2, Contact(str1,s))
   ∧ TRUE(t1, t2, Contact(str2,s))
   ∧ TRUE(t1, t2, (A[right_end(str1)] = A[location(s)]))
   ∧ TRUE(t1, t2, (A[left_end(str2)] = A[location(s)]))
   ∧ TRUE(t1, t2, Continuity(str1,str2))
   ∧ TRUE(t1, t2, (Am[right_end_velocity(str1)] > ZERO))
   ∧ TRUE(t1, t2, Direction_of(UPWARD, right_end_velocity(str1))))
  ⇒ (∃ t3 ∈ temporal_term, t3 > t2,
     TRUE(t2, t3, ¬Contact(str1,s))
     ∧ TRUE(t2, t3, ¬Contact(str2,s))))
```

These two rules describe two different situations where a contact can cease: a taut string sliding (in either direction) and reaching its physical end, and a double catenary being pulled at one end and thus rising up to get detached from the middle support. In both cases the ceasing Contact relation will be removed from the current set of active relations, and the view structure for the given scenario will be computed again: in the latter case, for instance, this will lead to the birth of a new catenary from the disappearance of the two catenaries involving the Contact relation above.

In the second rule we can see an example of how the metric diagram is used: determining the direction of a velocity typically requires numerical information (the coordinates of the velocity), which can be attained from the metric diagram.

As regards the occurrence of a new contact relationship, the problem is more difficult. The main reason is that a physical string is divided into segments accordingly to its current contacts with the supports, hence the current decomposition doesn't reflect the new contact that might occur because of a string motion. This fact makes the formulation of the rules more complex, and is reflected in a heavier dependence of the rules on the numerical information contained in the metric diagram.

Consider, for example, the following rule, which describes a string segment coming into contact with a support:

```
(FOREACH str ∈ string_segment, FOREACH s ∈ slide_support,
 FOREACH t1, t2 ∈ temporal_term,
  (TRUE(t1,t2,¬Contact(str,s))
   ∧ (FOREACH t3, t1 ≦ t3 < t2,
      TRUE(t1, t3, Greater_than_zero(Distance(str,s))
       ∧ TRUE(t1, t3, Less_than_zero(Derivative(Distance(str,s)))))
   ∧ TRUE(t2, t2, ¬Greater_than_zero(Distance(str,s)))
  ⇒ (∃ t4 ∈ temporal_term, t4 > t2,
     TRUE(t2, t4, Contact(str,s))))
```

It is apparent that this rule heavily rests on the metric diagram, in that the various conditions on the distance can be determined only by means of quantitative information. Notice that a zero distance between a string segment and a support implies only a geometric contact between the two entities, and not necessarily a physical contact; therefore in the rule above we specify that the zero distance condition must be reached with a velocity greater than zero.

The possible connectivity changes specified by the set of rules for a given view and process structure must be added to the set of limit hypotheses found for that structure after the limit analysis [for84]. This will produce a description of (hopefully) all the ways a situation can be modified by the active processes, with possible ambiguities concerning which change actually occurs first. A detailed discussion of this point is out of the scope of this paper.

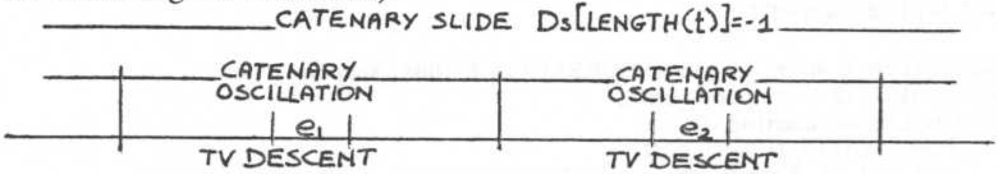
4. Temporal reasoning and QPT.

In qualitative causal reasoning, time often may be simply seen as an independent quantity, over which all other quantities evolve (i. e. change their values). However there are situations where this simple vision of time is somewhat inadequate; on the contrary, time should explicitly appear as a quantity in the model considered, possibly linked to other quantities by appropriate relations (e. g. qualitative proportionalities). In such cases , beside predicting the future changes occurring in a given scenario, usually we want to draw some inferences about the time taken by these changes to happen.

Forbus[for84] proposed a solution for this problem by introducing, for an episode in a history, the notions of rate, duration and distance, along with the relation holding (under appropriate hypotheses) among them. Here we present a possible extension of those concepts, which allows more temporal inferences to be drawn about an envisioned behavior. In particular a new entity, the **Temporal view (TV)** is added to the existing theory in order to describe the occurrence of a certain episode within the history of a given situation, and state the possible relationships existing between the various quantities involved in the episode and its duration.

The structure of a TV resembles the structures of the other QPT entities, in that it comprises the usual fields: individuals, preconditions, quantity conditions and relations. These fields have the same meaning as the corresponding fields in processes and individual views, except that time is explicitly mentioned in the form of episodes; for example, the preconditions comprise constraints on the start and end instants of a particular episode mentioned among the individuals, and the relations field includes relations linking the duration of an episode with other quantities of the individuals involved in the TV.

Let's consider, for example, the following situation: an oscillating catenary is pulled at one end so that the length of the string segment is decreasing. The envisionment for this situation leads to the history depicted below (for the sake of brevity we have left out the process describing the oscillation).



In this case we can consider an episode corresponding to the bottom of the catenary moving downward between two heights h1 and h2; this episode can be described by the following TV:

Temporal_view **Descent(t,s1,s2,h1,h2,e)**
Individuals:
    t a string_segment;
    s1, s2 a slide_support or a block_support;
    e an episode;
Quantity_conditions:
    TRUE(start(e), end(e), Catenary(t,s1,s2));
    TRUE(start(e), end(e), Catenary_oscillation(t,s1,s2));
    TRUE(start(e), start(e), (M A[hmin(Catenary(t,s1,s2))] = h1));
    TRUE(end(e), end(e), (M A[hmin(Catenary(t,s1,s2))] = h2));
Relations:
    Let average_speed be a quantity
    A[average_speed] = (M A[bottom_speed(Catenary(t,s1,s2))] start(e)) +
        (M A[bottom_speed(Catenary(t,s1,s2))] end(e));
    Ds[average_speed] = ZERO;
    duration(e) $\propto_{q-}$ average_speed;

Notice that the TV parameters include not only the individuals involved, but also an episode and a couple of height values defining the episode itself. Introducing explicitly the description of an episode occurrence allows us to specify, by means of the relation field, what is known about the episode duration. In our particular case, we can say that the duration is inversely proportional to the average catenary speed (which corresponds roughly to the above mentioned relation linking rate, distance and duration, with the speed playing the role of rate).

Let's now consider two different instances of our TV in the history for the oscillating and sliding catenary (see above). In general, the only inference we can draw about the relative durations of the two episodes involved in these instances comes from the above mentioned relation between duration and speed (provided we know the mutual relation between the average speeds). However, in this case, we know that the length of the string segment is decreasing because of the catenary slide process, and thus we can say that the duration of the second episode will tend to be less than the duration of the first. In fact the catenary is comparable with a pendulum, whose period is known to be proportional to its length; moreover, a shorter length of the string segment corresponds to a shorter length of the pendulum, which leads to a briefer period.

We can represent this knowledge by introducing a new TV, which imposes the appropriate relation between the episode durations when two instances of the Descent TV occurs in coincidence with a (monotonically) changing length string.

Temporal_view **Oscillation_damping(t,s1,s2,h1,h2,e1,e2)**
Individuals:
    t a string_segment;
    s1, s2 a slide_support or a block_support;
    e1, e2 an episode;
Preconditions:
    (start(e2) >= end(e1));
Quantity_conditions:
    Descent(t,s1,s2,h1,e1);
    Descent(t,s1,s2,h1,e2);
    TRUE(start(e1), end(e2), Catenary_slide(t,s1,s2));
    TRUE(start(e1), end(e2), (Dm[length(t)] > ZERO));
Relations:
    (duration(e2) − duration(e1)) $\propto_{q+}$ ((M A[length(t)] end(e2)) −
        (M A[length(t)] start(e1)));
    Correspondence(((duration(e2) − duration(e1)),ZERO),
        (((M A[length(t)] end(e2)) − (M A[length(t)] start(e1))),ZERO);

It is apparent that the qualitative proportionality and the correspondence written in the relations field define the correct relationship between the episode durations and the direction of (monotonic) change of the string length. It is worth noticing that the inference we can draw from this TV (i. e. the ordering relationship between the durations) is not based on the relation linking duration, rate and distance, but it comes from the knowledge we have of the occurring behavior; the purpose of the TV concept is making it possible to provide a domain model with this kind of knowledge.

5. Discussion

We have presented a framework for qualitative reasoning about flexible objects, including dynamic, kinematic and temporal aspects. Although still in its early stage, we trust that the development of this work will eventually leads to interesting results for potential applications (e.g. manipulator−performed cable placing). Two open points are reported here as an indication for future work.

The Locality Problem

Although strings are most carefully described using a *local* model (e.g. molecular [gam85]), global inferences about the behavior of certain configurations seems natural for us and can be drawn by *global* models. For instance, a molecular representation can predict exactly which part of a moving string is about to collide against an obstacle, while a global model will in general predict an ambiguous collision. On the other hand, ambiguity is a consequence of the particular kind of discretization we choose. In this sense, a global prediction is rather a set of options which delimits the possible behaviours [de85]. Adding details to the global representation is a problem of graceful extension, which means here improving the MD/PV model adopted, insufficient per se to draw inferences about the motion of the string itself.

Temporal Reasoning

Many quantities are defined with reference to timing concepts. A conclusion about the trend of the durations of a class of episodes enables further conclusions about the evolution of other related quantities (e. g. speed, acceleration, and so on). A better understanding of the use of time in qualitative relations is necessary. Also the problem of merging the landmarks coming from limit analysis with those deriving from prerequisites analysis must be further investigated.

References

[all83]

J. F. Allen, Maintaining knowledge about temporal intervals, *Comm. ACM 26*, (1983), .

[de85]

J. de Kleer and J. S. Brown, A Qualitative Physics Based on Confluences, in *Formal Theories of the Commonsense World*, J. R. Hobbs and R. C. Moore (ed.), 1985, Ablex.

[di87]

M. di Manzo and E. Trucco, Commonsense Reasoning about Flexible Objects: a Case Study, *Proceedings of AISB−87*, Edimburgh, Apr. 1987.

[di88]

M. di Manzo, E. Trucco and D. Tezza, Commonsense Strings, Climbers and the Qualitative Catapult, Int. Report, University of Ancona, Ancona, Italy, 1988.

[for83]

K. Forbus, Qualitative Reasoning about space and motion, in *Mental Models*, D. Gentner and A. S. Stevens (ed.), Hillsdale, N.J., 1983, Erlbaum.

[for84]

K. Forbus, Qualitative proces theory, *Artificial Intelligence 24*, (Dec. 1984), .

[for87]

K. D. Forbus, P. Nielsen and B. Faltings, Qualitative Kinematics: A Framework, *Proceedimgs of IJCAI−87*, Milan, Aug. 1987.

[gam85]

L. Gambardella, F. Gardin and B. Meltzer, Analogical Representations in Reasoning Systems, JRC EURATOM AI Lab. Int. Report, 1985.

[kui84]

B. Kuipers, Common sense Causality: Deriving Behavior from Structure, *Artificial Intelligence 24*, (1984), .

[kui]

B. Kuipers, Modelling Spatial Knowledge, *Proceedimgs of IJCAI−77*, , Morgan Kauffman.

[mcd82]

D. McDermott, A Temporal Logic for Reasoning about Processes and Plans, *Cognitive Science 6*, (1982), .

[sho85]

Y. Shoham, Naive Kinematics: one aspect of shape, *Proceedimgs of IJCAI−85*, Los Angeles, Aug. 1985.

[sho87]

Y. Shoham, Temporal Logics in AI: Semantical and Ontological Considerations, *Artificial Intelligence 33*, (Sep. 1987), North Holland.