Simulating Both Dynamic and Kinematic Behaviors of Mechanisms

Pearl Pu
University of Connecticut
Department of CS and Engineering
U-155 Storrs, CT 06268
email: pu@carcvax.uconn.edu

19 February 1989

1 Problems

Humans, including children [7], can describe mechanisms using whatever knowledge and information they have. That is they often switch between dynamic and kinematic descriptions depending on the information available to them. Furthermore, they seem to perform such reasonings subconsciously. For instance, they can describe how meshed gear trains work by just addressing how rotational velocities are transferred into velocities of opposite directions. Sometimes, however, they analyze the motion propagation starting with the torque source and how the torque brings motion to the rest of the system.

One of the goals in qualitative physics is to achieve a cognitively more plausible account of how various physical systems behave so that both commonsense and more complex physical principles can be captured. Theories in this field, however, tend to draw a clear line between the dynamic and kinematic behaviors of physical devices, thus deviating away from how humans normally describe mechanisms. In particular, dynamic theories cannot be applied to produce kinematic behaviors. Neither can kinematic theories be applied to deal with dynamic devices. Our key observation is that humans use motion propagation as an unified way of accommodating both dynamic and kinematic descriptions.

KREATOR [9] is constructed to combine the qualitative simulation of both dynamic and kinematic behaviors of a device based on the notion of force/velocity propagation. Whether the system's description of the mechanism is dynamic or kinematic depends on the input. Thus we can view the system as taking whatever information is available and trying to use that knowledge the best it can to produce an answer about the behavior of the mechanism. Due to the page limitations of this paper, we can only briefly describe the major aspects of this approach.

2 A Broad Perspective of KREATOR

Besides aiming at constructing an unified approach to describe both dynamic and kinematic behaviors of mechanisms, several other related unsolved issues are also investigated in this work:

- One can relatively easily simulate a mechanism if everything is quite normal. But if one of the parts of a mechanism becomes suddenly obstructed, how can the system react to this situation?
- According to Schwarm [10], "a *mechanism* is a combination of rigid bodies so arranged that the motion of one compels the motion of the others, according to a law depending on the nature of the combination." Motion compelling among objects is being viewed as a way of propagating causalities. But can there be a computational model that formalizes motion propagation? Is it force being propagated? Or is it velocity? Or is it something else?
- For a while, we didn't know whether a pure qualitative kinematic theory could be constructed. Now Faltings [4] and Joskowicz [5] have shown that we must use a quantitative model for the underlying geometry and extract qualitative information from that quantitative model. These solutions give us the confidence to model the geometry of any mechanism of planar motion and establish a qualitative model. Furthermore, if the underlying geometry changes, for instance a part is being deleted, a modification in the qualitative model shouldn't be too hard to achieve. But current theories in the field never dealt with simulating a device with a component suddenly deleted. Can this problem be solved?

As it turns out, our approach offers advantages in solving the above mentioned problems. We will emphasize in this paper how we construct a computational model for the notion of motion propagation. This paradigm is generalized to treat three types of mechanical behaviors: dynamic, oscillatory, and kinematic motions. In the end we will briefly sketch some preliminaries and background materials used in the work.

3 A Computational Model for Motion Propagation

In descriptions such as "force imparts from one object to another," "momentum transfers from one object to another," and "the velocity of one object propagates to another," we observe one unique characteristic — they all employ the notion of propagation of some physical entities such as force, momentum, or velocity. Our surveys [1] and [2] show that this style of reasoning about the interactions (or behaviors) among objects in a mechanism has been used effectively to describe how mechanisms work in encyclopedias, text books, and repair manuals. When human experts teach the novices, their oral descriptions also take such a form of propagation reasoning. However, the notion of motion propagation has not been emphasized in the field of qualitative simulation.

We now describe a new model to perform the qualitative simulation of mechanisms based on the notion of motion propagation. There are certainly many ways to represent motion propagation — force transfer, velocity transfer, momentum transfer, or even energy transfer. We decided to use force and velocity as a pair to represent motion propagation. The product of force and velocity is the power variable. So power transfer can also be represented. Furthermore, both displacement and momentum can be calculated from the values of velocity and force respectively by taking intergration with respect to time.

3.1 Constraints in a Mechanism

The question of constraints in a mechanism must be addressed before we outline the method to propagate force and velocity. When objects are assembled to form a device, geometrical and physical constraints will be introduced as a result. We use the bond graph method [6] to express such constraints. The simulation of the behavior of the device relies on the constraints to propagate effects from the force or velocity sources to every object in the device.

To convert a device from its physical model to bond graph notation, we can follow a step-by-step procedure, which is outlined in detail in Pu [9]. This process involves two steps for every device: 1) identifying the component and connection types in the device, and 2) putting together the components and connections in a constraint according to several rules.

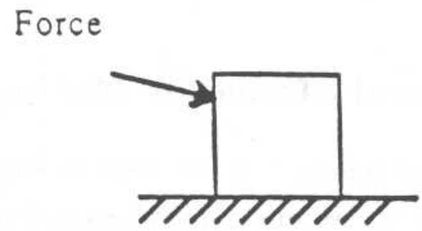

Figure 1: A block on a smooth surface

3.2 Force and Velocity Flows

To define the force/velocity flow for a particular component, consider a block resting on a smooth surface as shown in Figure 1. If a force is applied to the block for a period of time and if the block is not obstructed in any way, it will accelerate and pick up some speed as a result. If, however, the block is obstructed, then the block will not move even though there is an acting force. Some analogies can be made in this style of reasoning: the force applied to the object can be viewed as the input to this simple mechanism, and the velocity picked up by the object can be viewed as the output. Between the input and output, there is a latch which either allows or disallows the flow to propagate from the input port to the output port. Using causal notions, we can also view force as the cause and velocity as the effect. Thus we are proposing the flow propagation model for components as shown in Figure 2. The arrows in the diagram refer to the input and output rather than the directions of the applied force and the resulting velocity. Viewed differently this diagram can be also interpreted as the cause, effect, and enablement/disablement of motion as shown in Figure 3.

For the case of force/velocity passing across a connection, consider a transformer such as a lever as shown in Figure 4. When a force is applied to the left, there will be a force of opposite sign coming out at the other end. This suggests that we ought to view motion propagation across a connection in terms of in-coming force and out-going force. The out-going force will in turn serve as an input force to whatever components that are immediately connected to this connection. What about velocity flows? In a connection, velocity goes in and comes out just like the force flow, except

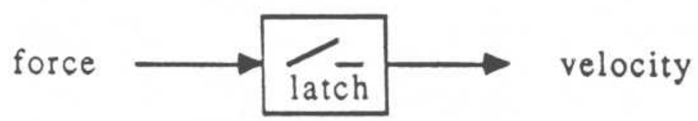

Figure 2: Force and velocity flow for components

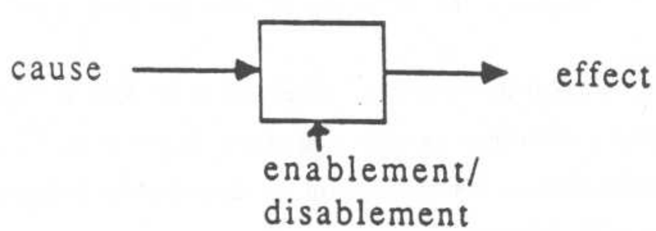

Figure 3: Cause and effect of motion

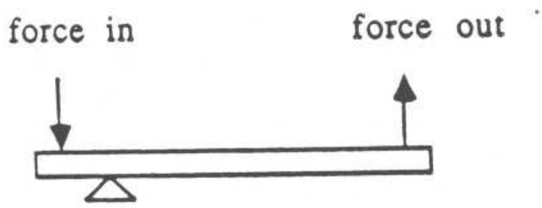

Figure 4: A lever

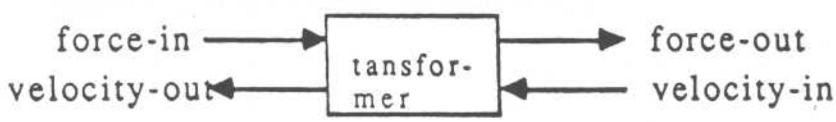

Figure 5: The force and velocity flow for a transformer

the direction of the flows is in the opposite direction of the force's. Thus we propose the force and velocity flow for the transformers as what is shown in Figure 5. In Pu [9], we discuss in detail how values will be propagation in connections and we show that for different types of connections, the flow diagram is slightly different.

In the next few sections, we describe how this force/velocity propagation paradigm can be used to model the dynamic, oscillatory, and kinematic motions respectively.

3.3 The Dynamic Case

When there are multiple components in a device connected in a particular way, if we know the external force source(s) of the system, the question of computing the force exerting on every com-

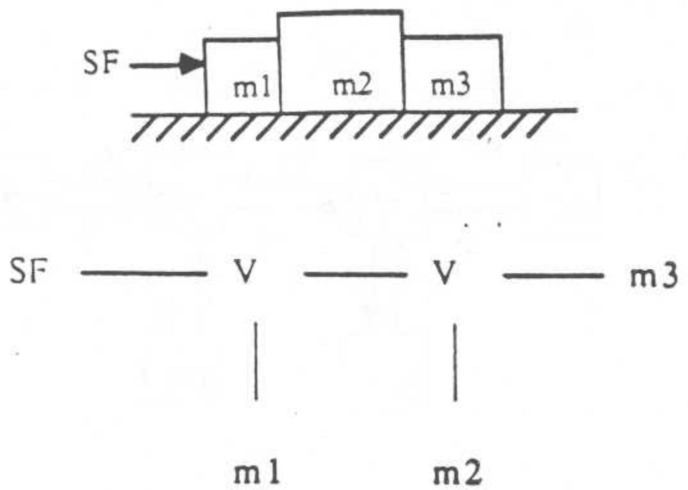

Figure 6: 1) A Series of blocks 2) Blocks in bond graph

ponent and calculating the velocity each component is to carry is a dynamic problem. To derive the solution, consider the following example.

Several blocks are arranged on a smooth surface as shown in Figure 6. An external force, SF [1], is applied to the block on the far left. The question is that what will the velocity of each of the blocks be after a small period of time?

Using the bond graph representation, the constraint graph will look like what is shown in part 2 of Figure 6. The symbol I represents inertial objects and V for common-velocity junctions. (If two objects are physically bound so that they must travel with the same speed, they share a common-velocity junction as their constraint.)

According to our notion of force and velocity flow, the forces flow into the components and the velocities flow out from these components. For connections, on the other hand, forces flow *across* (or through) them in one direction and velocities flow across them in the opposite direction. Putting the pieces together, we obtain the force and velocity flow diagram as shown in Figure 7

In this diagram, we only know the direction and magnitude of SF. The process of finding out the distribution of SF to each of the blocks is the job of our simulator. After the simulator obtains the force distribution, it should then calculate the velocity for each block.

[1] SF denotes a force source. Refer to Pu [9] for the complete set of symbols.

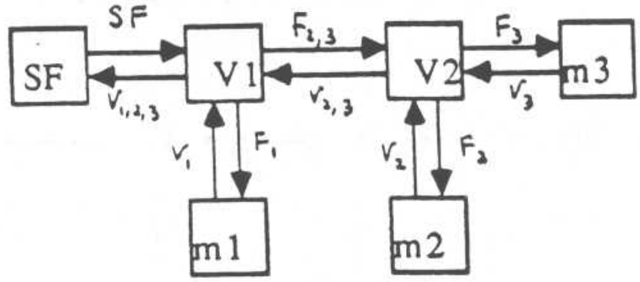

Figure 7: The force and velocity flow directions for the three-block example

This model also proposes to reason about the dynamic behavior of a compound device by breaking down the problem into subproblems and then tackling each of the subproblems separately. Notice that suppose we know the distribution of SF to F1, F2, and F3. The problem then becomes the behavioral reasoning of three individual blocks, each of them being supplied with an external force. We can now easily apply Newton's first law to each case (all three cases are the same): since force will cause an acceleration, therefore the block's velocity will change from zero to positive. (Assume the force is applied in the positive directions and the blocks are initially stationary.) Once we know the velocities of individual blocks, we can then propagate them backward so that some synthesis for global behavior can be performed. Specifically in this example, the synthesis rule states that the velocities of all blocks have to be consistent, since they are joined by common-velocity connections.

3.4 The Oscillatory Case

A good way to characterize oscillatory behavior is by realizing systems that exhibit such behaviors conserves energy in ideal cases.

Assuming that energy does not dissipate in these systems, how do we apply the force and velocity model to construct the simulation algorithm for such cases? Notice that in an energy-conserved system, force and velocity flows also exist. It differs from the dynamic case only in the way the input is specified. In dynamic systems, the input is expressed as a function of time. So the SF changes as time goes on. In energy-conserved systems, there is no input.

We propose a loop propagation method for the energy-conserved systems. That is, when the force values are propagated from some nodes[2] to the rest of the nodes in a device, the flow turns around and propagates back the velocity information. When it is done, it turns around again and propagates force information again.

3.5 The Kinematic Case

How do we apply the force and velocity flow model to the kinematic case? We propose to propagate the velocity sources from those components that receive them (the initial nodes) to every other node in the system. When a component is encountered, the simulator sets the velocity of that component to the input velocity. When a connection is encountered, the simulator uses the kinematic rules to find out the output velocity from the input velocity in that connection. Continuing this way, the simulator will eventually reach every node in the constraint graph and that is the end of one cycle for the kinematic simulation. Since the velocity sources are also given as functions of time, we go back to the initial nodes and perform another simulation of the device with some new inputs.

3.6 Algorithms

We outline the three force and velocity propagation algorithms for the case of dynamic, energy-conserved (those systems that produce oscillatory behaviors), and kinematic systems respectively.

- forward dynamic systems — the simulation is to follow the force flow from external force sources (initial nodes) across every connection and calculate how that external force is distributed to all components in the system; then it is to calculate the velocity for each component. It repeats this pattern continuously with new force inputs each time.

- energy conserved Systems — the simulator is to follow the force flow from the initial nodes to the final nodes and calculate the force distribution at the same time, then it is to follow the velocity flow from final nodes to the initial nodes. It repeats this pattern.

- kinematic systems — the simulator is to follow the velocity flow from the initial nodes to the final nodes and calculate the velocity distribution for each component. It repeats this pattern continuously with new velocity inputs each time.

[2] Notice that there is no initial nodes in energy-conserved systems. Initial nodes are arbitrarily chosen.

4 Propagation Rules

The propagation flow models for the three types of behaviors have been outlined in the previous sections. We now briefly sketch our approach to calculate the distribution of force and velocity, etc.

For components, the problem is to calculate the changes in velocity given force as an input. For capacitors, such as springs, the problem is to calculate the changes in force given velocity as an input. This is done by using deKleer's confluence model for capacitors, inertial objects, and resistors. For connections, we have two sets of rules. One set handles how forces will be distributed for each of the connection types. The other set handles how velocities are distributed for each of the connection types.

Again whether the simulation is to produce dynamic, oscillatory, or kinematic behaviors depends on input information. The propagation rules just described reside with the particular entity that it's associated with.

5 Conclusion

It is important that theories in Qualitative Physics can use the same model to reason about both dynamic and kinematic behaviors of mechanisms, thus utilizing the available information. Using the current results from Faltings and Joskowicz, we have shown that such a model based on the propagation of force and velocity exits. Furthermore, it produces dynamic, oscillatory, and kinematic descriptions for a wide variety of mechanisms including intermittent devices. With some synthesis rules, our system can also reason about inconsistencies in mechanism, such as the three interlocking meshed-gear mechanism.

5.1 Position Treatment

We wish to point out that KREATOR doesn't treat position in a quantitative sense directly, since we have set the initial goal to achieve a pure symbolic reasoning of mechanisms. To incorporate position, either Faltings or Joskowicz's works can be used to generate the necessary qualitative information, such as the kinematic state, as the input to our system. So the kinematic description generated by our system is largely associated with velocity, its propagation in the mechanism, and how the mechanism goes through kinematic states.

5.2 Simulation Output

To illustrate simulation outputs, works by Nielsen [8] and deKleer [3] produce envisionment diagrams which are networks of nodes. Each node is a possible state of the mechanism. Links in the network correspond to possible state transitions.

A major difference between our results and theirs is that we not only produce the state-to-state transition diagrams, but also object-to-object transition diagrams. This latter diagram corresponds to how motion is actually propagated from the sources to every objects in the mechanism. The simulation thus produces a trace of causality for mechanisms as long as they don't have motion loops (perpetual motions are not yet considered with the model).

A The Background

A.1 Device-based Model and Beyond

The underlying ontology of this work adopts that of deKleers [3] device-based model. In particular, a device consists of components and connections. Rather than just assuming connections as links between a pair of two components, we use a representational entity called the connection frame to capture the geometrical structure of the connection, and the dynamic and kinematic behaviors of the connection.

A.2 Representation of Primitives

Since we are not so much concerned about recognizing whether an object is a spring, a block, etc., we have constructed a set of abstract objects to accommodate all possible mechanical objects. The recognition process as mentioned can be solved by methods such as those suggested by, for instance, Faltings [4] and Joskowicz [5]. In our system, all objects belong to one of the abstract categories. Because these categories are defined at a very abstract level, we do not suffer from problems associated with enumerating a set of mechanical objects at a lower level, such as the approach taken by Stanfill [11].

The set of primitive components and connections accountable in our model is:

- inertial objects
- springs

- resistors
- transformers
- 2-port common velocity junctions
- 2-port common force junctions

The 2-port common velocity and force junctions can be generalized to the case of n-port, where n is any positive number greater than or equal to 2.

A.3 The [+,0,−] Calculus

As was pointed out in Struss [12], the [+,0,−] calculus has many drawbacks as a quantity representation for real numbers. However, our model is not tightly-coupled with any particular quantity representation scheme, nor is it relying on the calculus provided by deKleer [3] for the calculation of quantities. In fact, as suggested in Pu [9], if a proportionality reasoning module based on comparative analysis [13] is incorporated into the simulator, the system improves on its quantitative analysis. However, for analysis and illustrational purposes, we have assumed the [+,0,−] calculus for this paper.

References

[1] *The Way Things Work: An Illustrated Encyclopedia of ctechnology.* Simon and Schuster, New York, 1971.

[2] John Bickford. *Mechanisms for Intermittent Motion.* Industrial Press Inc., New York, 1972.

[3] Johan deKleer. A qualitative physics based on confluences. *Artificial Intelligence*, 24, 1984.

[4] Boi Faltings. *Qualitative Place Vocabularies for Mechanisms in Configuration Space.* Technical Report UIUCDCS-R-87-1360, Ph.D thesis, University of Illinois at Urbana-Champaign, 1987.

[5] Leo Joskowicz. *Reasoning about Shape and Kinematic Function in Mechanical Devices.* Technical Report, Ph.D thesis, New York University, 1988.

[6] Dean Karnopp and Ronald Rosenberg. *System Dynamics: A Unified Approach.* John Wiley & Sons, New York, 1975.

[7] Kathleen Metz. The development of children's problem solving in a gears task: a problem space perspective. *Cognitive Science*, 24, 1985.

[8] Paul Nielsen. A qualitative approach to mechanical constraint. In *Proceedings of National Conference on AI*, August 1988.

[9] Pearl Pu. *Qualitative Simulation of Ordinary and Intermittent Mechanical Mechanisms*. Technical Report, Ph.D thesis, Computer Science Department, University of Pennsylvania, 1989.

[10] Peter Schwamb, Allyne L. Merrill, and Walter H. James. *Elements of Mechanism*. John Wiley & Sons, Inc., 1947.

[11] Craig Stanfill. *Form and Function: The Representation of Machines*. Technical Report TR-1347, Computer Science Department, University of Maryland, 1983.

[12] Peter Struss. Mathematical aspects of qualitative reasoning. *International Journal for Artificial Intelligence in Engineering*, 1988.

[13] Daniel S. Weld. Comparative analysis. *Artificial Intelligence*, 1988.