

Shape and Function in Mechanical Devices

Leo Joskowicz

Department of Computer Science
Courant Institute of Mathematical Sciences,
New York University
251 Mercer Street,
New York, NY 10012
joskowicz@nyu-csd2.arpa

ABSTRACT

This paper describes a two-step algorithm for the qualitative analysis of mechanical devices. The first step takes the geometrical description of the parts and their initial position and produces a description of the possible relative motions of pairs in contact by computing the configuration space of those pairs with respect to selected motions. Given the possible relative motions and an input motion, the second step computes the actual motion of each object for fixed axis mechanisms using a constraint propagation, label inferencing technique. The output is a state diagram describing the motion of each part in the mechanism.

Shape and Function in Mechanical Devices

By

Leo Joskowicz

Department of Computer Science
Courant Institute of Mathematical Sciences, New York University
251 Mercer Street, New York, NY 10012

There has been considerable attention in recent years on reasoning about the physical world. One of the areas that has been considered is reasoning about rigid objects. We are interested in reasoning about assemblies of rigid objects, in particular in the domain of mechanical devices. Among the reasoning tasks that involve mechanical devices, prediction of behavior is of particular importance. The problem of predicting the behavior of a mechanical device can be stated as follows: given a structural description of a device in terms of its parts and their shapes, the topological connections between the parts, and an input motion, determine the behavior of each of the parts of the mechanism. The behavior is expressed as the set of motions the parts of the mechanism can have.

Figure 1 shows an example of the type of mechanism we propose to analyze. The system consists of an input cylindrical gear *Gear1* that is mounted on a fixed axis. This gear meshes with another cylindrical gear, *Gear2*, that is also mounted on a fixed axis. *Gear2* is attached to a threaded rod that is in turn threaded through a nut, *Nut*. The nut is attached to the slider of a prismatic joint whose rack is fixed in space. A qualitative description of the behavior of this device runs as follows: When the input gear turns, the nut advances or retracts depending on the direction of the rotation of the input gear and the direction of the pitch of the threaded rod.

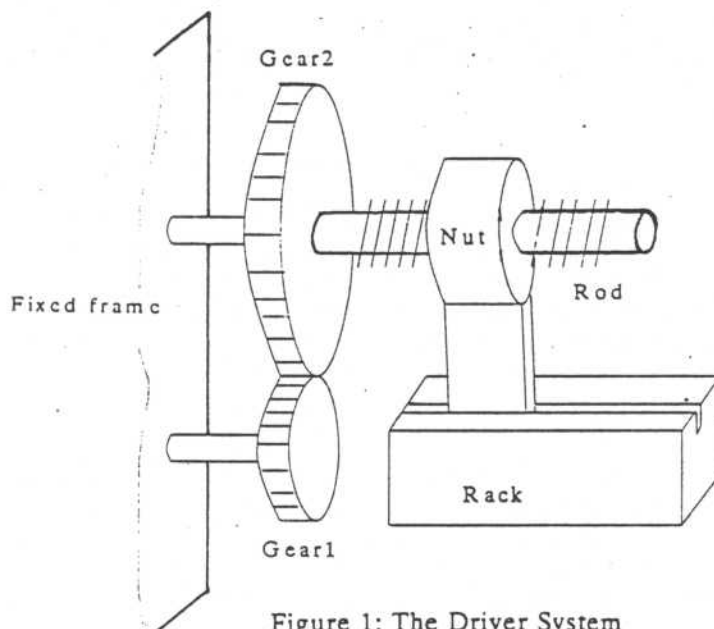


Figure 1: The Driver System

Previous work in qualitative physics has addressed the problem of determining the qualitative behavior of a physical device from its structure. In DeKleer and Brown's formalism [1], a device consists of three types of constituents: materials, components and conduits. Components are elementary parts that operate on and change materials. Conduits are components that do not change materials; they transport the material from one component to the other. Behavior is achieved by transporting materials from one component to the other through conduits. Applying this paradigm to the domain of mechanical devices amounts to considering motions as materials that are transported by mechanical parts and that are modified by pairs of parts. For example, in a train of gears, the components are gear pairs that modify the material "rotation",

transported by individual gears and axes. The function of the particular pair configuration is stored in the component description. Individual parts are not considered components. This is in contrast with the modeling of electrical devices as described in [1], where electrical parts (resistances, light bulbs, batteries) are components and their function does not change in different configurations (radios, heaters, etc.). A topological description of their connections is sufficient. For mechanical parts, the interaction between two gears is different from the interaction between a worm gear and a gear, leading to a different type of behavior. This requires that all the relations between two particular objects be defined in advance. Since the number of possible relations is very large and the geometry of the objects can vary, an exhaustive list of such relations is implausible. Forbus' Qualitative Process Theory suffers from the same limitations. A more general model that supports geometrical reasoning is required.

1. An algorithm for the analysis of mechanisms

We propose a two-step algorithm for the analysis of mechanisms. First, given a geometrical description of the objects and their initial positions, Local Interactions Analysis finds the possible relative motions of all pairs of objects that are in contact. Possible relative motions of objects in contact are expressed in terms of a small set of parametrized motion predicates (such as *rotates(A, axis(O), parameter)*, for object A), and a set of algebraic relations between parameters that indicate the dependencies between both motions.

The second step is the Global Interactions Analysis. Given the pairwise possible relative motions and an input motion, it determines the actual motion of each object. Two classes of mechanisms are distinguished here: fixed axis mechanisms and movable axis mechanisms. Fixed axis mechanisms are those mechanisms for which rotary axes do not move in space. Movable axis mechanisms have at least one rotary axis that moves in space. For fixed axis mechanisms, we build a constraint propagation network where each object is represented as a node, and each pairwise relation as a constraint edge between two objects. The initial motion is propagated as a label in the network, and nodes are labeled with possible motion predicates according to the pairwise relations. When the propagation halts, each node has a label that corresponds to the motion(s) of the object represented by the node. For movable axis mechanisms, we provide a heuristic rule to determine a lower bound on the degrees of freedom of the entire mechanism.

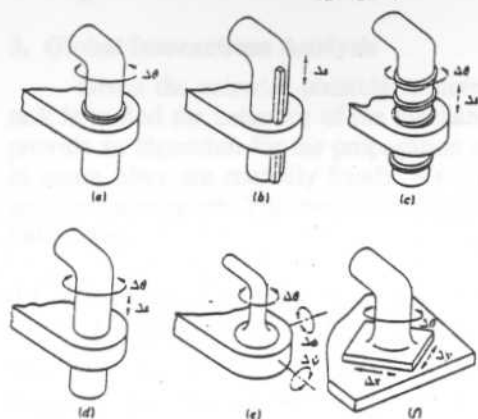
2. Local Interactions Analysis

We categorize pairs of objects in contact (kinematic pairs) following Reuleaux's classification [3]. Two objects in contact can form either a lower pair or a higher pair. Lower pairs are pairs in which the contact between the two objects takes place along a surface. Higher pairs are pairs in which the contact takes place along a line or a point. There are only six types of lower pairs, as illustrated in Figure 2, and infinitely many higher pairs. A typical higher pair is the pair formed by two meshing parallel gears. For each lower pair there is a simple motion predicate that describes the possible relative motions of the parts. For example, if A and B form a prismatic pair, this relation can be stated as:

$$\begin{aligned} \text{prism}(A, B, O) \iff & \\ & \text{translation}(A, O, Xa), \text{translation}(B, O, Xb) \\ & 0 \leq Xa + Xb \leq |\text{length}(A, O) - \text{length}(B, O)| \text{ along axis } O \end{aligned}$$

that is, A and B form a prismatic pair along an axis parallel to O. The possible relative motion of A is a translation along axis O by a distance Xa, and that of B is a translation along axis O by a distance Xb. The inequality involving Xa and Xb must always be satisfied where *length(A, O)* denotes the length of the prismatic section of A along axis O. The relations for *revolute(A, B, O)*, *helical(A, B, O)*, *cylindric(A, B, O)*, *spheric(A, B, point)* and *planar(A, B, plane)* are similarly defined.

To find whether two parts form a lower pair, we compute the configuration space of the translation and rotation of A with respect to B which is fixed, as defined in [4]. The configuration space of a moving object A with respect to a fixed object B is the set of all the positions of A such that A does not overlap with B. Figure 3 shows the configuration spaces for each one of the lower pairs. Two parts form a lower pair if their configuration space is one of the six configuration spaces shown in this figure. If the resulting configuration space is a point, then the two objects are attached.



The lower pairs

Pair	Symbol	Pair variable	Degrees of freedom	Relative motion
Revolute	R	$\Delta\theta$	1	Circular
Prism	P	Δs	1	Linear
Screw	S	$\Delta\theta$ or Δs	1	Helical
Cylinder	C	$\Delta\theta$ and Δs	2	Cylindric
Sphere	G	$\Delta\theta, \Delta\phi, \Delta\psi$	3	Spheric
Flat	F	$\Delta x, \Delta y, \Delta\theta$	3	Planar

(Taken from [5], pp 7-10).

Figure 2: The six lower pairs: (a) revolute, (b) prism, (c) helical, (d) cylindric, (e) spheric and (f) planar.

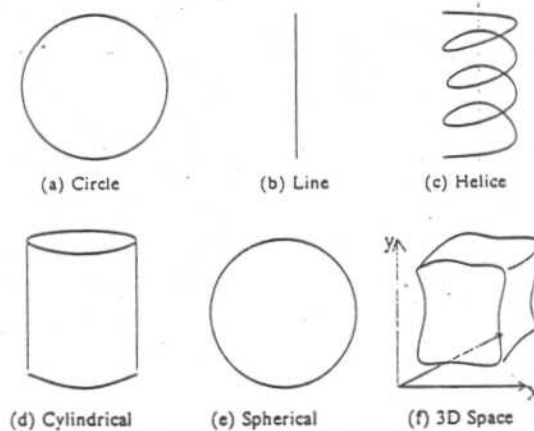


Figure 3: The configuration spaces for the lower pairs: (a) circle, (b) line, (c) helicoid, (d) cylindrical surface, (e) spherical surface, (f) 3D space.

Objects are assumed to be three dimensional objects whose edges are either line segments or arc segments and whose surfaces are either planes, cylinders or cones (or parts of them). Objects can be described as collections of cross sections along characteristic axes. A three dimensional object has six degrees of freedom in space and therefore its configuration space with respect to other fixed objects is 6-dimensional. Since it is impractical to compute the full 6-dimensional space, we compute the configuration space with respect to one degree of freedom at a time (rotation or translation) on selected two dimensional cross sections of the objects. The resulting configuration spaces will also be two dimensional. By taking unions and intersections of these configuration spaces we find the configuration space of the pair (this method is valid for all lower pairs except the helical one).

The recognition of higher pairs does not lend itself to a general method as the one described above. Two approaches are suggested here: a functional approach and a differential approach. In the functional approach, objects are described by properties. For example, a gear can be approximated by a cylinder with a number of properties such as the number of teeth, radius, etc:

cylindrical-gear(G) is defined by the properties
radius(G), number-of-teeth(G), size-of-teeth(G), axis-of-rotation(G), origin(G)
cylinder(G), belongs(origin(G), axis-of-rotation(G))

A higher pair is then described as a predicate that relates two functionally described objects. Two cylindrical gears that are mounted in parallel can be described by the following predicate:

parallel-gears(A, B)

Preconditions:

isa(cylindrical-gear(A)), isa(cylindrical-gear(B))
size-of-teeth(A) = size-of-teeth(B)
fixed(axis-of-rotation(A)), fixed(axis-of-rotation(B))
parallel(axis-of-rotation(A), axis-of-rotation(B))
distance(origin(A), origin(B)) = radius(A) + radius(B)

Then the relation between the possible motions is:

rotation(A, axis-of-rotation(A), origin(A), θ_A) IFF rotation(B, axis-of-rotation(B), origin(B), θ_B)
 $\theta_A = -\theta_B * \text{number-of-teeth(A)} / \text{number-of-teeth(B)}$

Two objects A and B form a parallel gear pair if they satisfy the preconditions.

For unknown higher pairs, several rules can be used to deduce the differential behavior of the two parts. Their integral behavior can then be deduced from their differential behavior. The analysis at the differential level consists of determining the behavior of the two parts at the next infinitesimal instant. The analysis at the integral level determines the behavior over a period of time. The predicates for lower and higher pairs shown before (*prism(A, B, O)* and *parallel-gears(A, B)*) describe both integral behaviors. It is possible to infer the differential and integral behavior of a higher pair using a set of differential behavior rules for solid objects. To illustrate how such an analysis can be made, suppose we want to infer the relation between two parallel gears. The following argument can be used: Let T_a and T_b be the two teeth in contact, where T_a belongs to gear A and T_b belongs to gear B. A rotation of gear A causes T_a to move along a circular path by a distance of dl . Since T_a is in contact with T_b and there are no obstacles that interfere with the motion of B, T_b will move along another circular path by the same distance dl . This constitutes the differential behavior of both gears. By determining how long T_a and T_b will be in contact, we can integrate this behavior. The behavior during an interval of time I is that B rotates together with A, in opposite directions by an angle of θ_A that equals $-\theta_B * \text{number-of-teeth(A)} / \text{number-of-teeth(B)}$. Let T_a' and T_b' be the two teeth following T_a and T_b in the direction of the motion of A and B respectively. Since T_a' and T_b' are part of A and B, they move with A and B respectively. Therefore they will be in contact before T_a and T_b stop being in contact (assuming the spacing between teeth is such that this is true). Another integration of behavior can now be based on symmetry arguments; B will turn when A turns in opposite directions. The same angle relationship as described above will hold for any time interval I . We thus obtain the relation for parallel gears. This argument is made more precise by using a set of rules that support this deduction. An example of such a rule is the differential Contact Rule:

Contact Rule: Transmission of force between two planar surfaces; let S_1, S_2 be two planar surfaces of two distinct objects O_1 and O_2 in contact. Let N_1, N_2 be the two normals to the point (or surface) of contact of S_1 and S_2 respectively. Then if a force is applied to O_1 , it will be transmitted to O_2 in the direction of the normal to the point (or surface) of contact, provided that O_1 can move in the direction of the force (assuming no obstacles and forces greater than friction).

Using this set of rules, with additional geometrical reasoning, we showed how to deduce the behavior of a worm gear meshed with a cylindrical gear. Although this method is not general, it can be used in some simple cases, especially in the domain of gears.

Kinematic pairs can be either simple or complex. Simple pairs are the ones described above i.e. those who have a single state corresponding to a single relative qualitative behavior. A complex pair is a pair that is described by several simple pairs, each corresponding to a different relative qualitative behavior. Each relative qualitative behavior is represented by a local state. A local state is created by a change in the contact points or surfaces between the two parts. Each new local topology is analyzed as a simple pair, and the transitions between states are conditions on the positional parameters of the objects. The resulting collection of states and transitions is called the local state diagram for the kinematic pair A, B.

The output of the Local Interactions Analysis is a set of local state diagrams containing relative motion predicates, one for each pair of objects originally in contact. This description corresponds to a functional description of the kinematic pairs.

3. Global Interactions Analysis

Given the pairwise possible relative motions and an input motion, the Global Interactions Analysis task is to find the behavior of the mechanism in terms of the motions of its individual parts. We will first provide an algorithm for the propagation of motion for mechanisms in which axes of rotation do not move in space (they are spatially fixed). We will then analyze the criteria necessary for mechanisms in which axes move in space. The mechanism in Figure 1 is an example of a mechanism in which all axes are spatially fixed.

3.1. Motion propagation Algorithm for fixed axis mechanisms

We will first consider mechanisms for which each pair has a single local state (simple pairs), and whose global topology does not change as the parts move. The problem of determining the motions of all the parts of a mechanism, given an initial input motion, can be viewed as a constraint propagation, label inferencing problem. Given a set of n terms with initial labelings and a set of constraints relating the terms, find a final term labeling that is consistent with the constraints. A constraint network for the Global Analysis is built by having one node for each part in the mechanism. The labels are the possible parametrized motions of the part (rotation, translation, fixed, undetermined, etc.) and their constraints. The constraints are the relations between pairs of objects found in the Local Interaction Analysis (*parallel-gears*(A, B), *prism*(A, B), etc.). A dummy node is introduced to represent the "source" of the input motion. All nodes are initially labeled "fixed", "fixed-axis" or "undetermined" except for the input node which is labeled with the initial motion. Figure 4 shows the constraint network for the mechanism in Figure 1. The input motion is propagated by starting at the input node, and examining all its successors. The new labeling of a successor is determined by intersecting the label found in it (including its bounds) with the possible motion of that object, as found in the constraint. In the example of Figure 4, the input motion is *rotation*(I, O, θ), the initial labeling of *Gear1* is *fixed-axis*(*Gear1*, O), and the relation between *I* and *Gear1* is *attached*(*I*, *Gear1*) (i.e. *motion*(A) \Leftrightarrow *motion*(*Gear1*)). Since *motion*(I) is *rotation*(I, O, θ) and *motion*(*Gear1*) is *rotation*(*Gear1*, O, θ), the intersection of *rotation*(*Gear1*, O, θ) and *fixed-axis*(*Gear1*, O) yields the new label, *rotation*(*Gear1*, O, θ) for *Gear1*. The intersection between two possible motions is defined in intersection rules such as: let L be the label of an object and *rotation*(A, O, θ_A) its possible motion. Then,

$$\begin{aligned} \text{rotation}(A, O, \theta_A) \cap L = \\ \text{if } L = \text{rotation}(A, O', \theta_{A'}) \text{ or } \text{fixed-axis}(A, O') \text{ and } O = O' \text{ then} \\ \quad \text{rotation}(A, O, \theta_A) \text{ where } \text{restrictions}(\theta_A) = \text{restrictions}(\theta_{A'}) \cap \text{restrictions}(\theta_{A'}) \\ \text{else } \emptyset \\ \text{if } L = \text{translation}(A, O', X_A), \text{ then } \emptyset \\ \text{if } L = \text{undetermined}(A), \text{ then } \text{rotation}(A, O, \theta_A) \text{ and } \text{restrictions}(\theta_A) \\ \text{if } L = \text{fixed}(A), \text{ then } \text{fixed}(A) \end{aligned}$$

The algorithm propagates the motion in a Breadth First Search manner to all nodes. If a label modification occurs for a node, the node and all its neighbors are added to a list of nodes to be updated. The algorithm stops when this list is empty, i.e. when the node labels cannot be modified any further. For each part, the label represents the possible motion of the object and its relation (via parameters) to the motion of the neighboring parts. The output is a single global state that contains the behavior of each part. This algorithm has been implemented in Franz Lisp.

The algorithm can be extended to deal with mechanisms that have complex pairs by building a set of global states consisting of the cross product of local states. Transitions between global states are constructed as the combination of the local transitions. To find the behavior of each part in a global state, each global state is analyzed using the algorithm described above. Some global states and transitions will be detected as infeasible, and thus be deleted from the global state graph.

Different global states can also be produced by changes in the topology of the mechanism when new contacts between parts are created or when old ones disappear. After the motion propagation algorithm has been executed, topological changes can be detected by computing the motion envelope (all the positions in space that an object occupies while moving) of each part. If two or more motion envelopes intersect, a new contact is created. This means that the bounds of the motions of the parts must be updated by propagation in the constraint network. The new contacts are analyzed locally as new pairs are created and a new

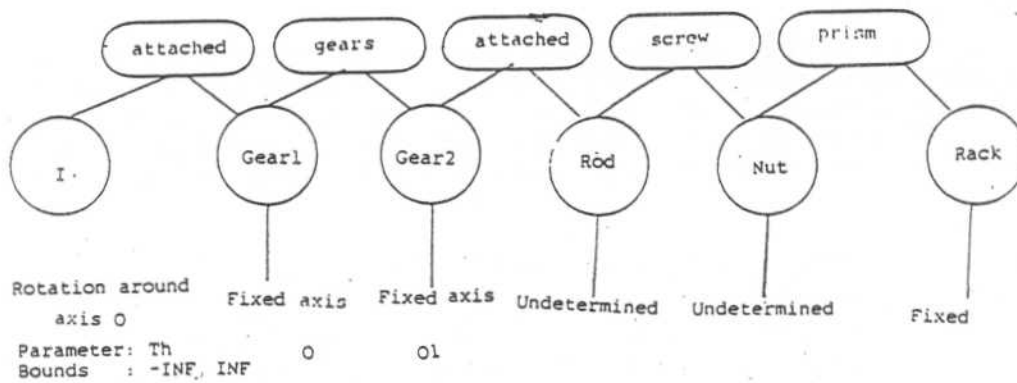


Figure 4: The initial constraint network for the Driver System.

constraint graph is built to correspond to the new global state. The transition between the current global state and the new one is specified as a condition on the positional parameters of the objects that came into contact. The resulting global state diagram is similar to the state diagram produced in [1] and to the graph of transitions in [2] used to explain the behavior of a physical system.

3.2. Mechanisms with movable axes

The algorithm described in the previous section cannot be generalized for mechanisms with movable axes since the combination of two simple motions (rotation, translation, etc.) can result in a complex motion. Nevertheless, the algorithm can still be applied to the parts of the mechanism with fixed axis, isolating the parts that have movable axes. A possible criteria for movable axes is the Kutzbach criteria [5], originally developed to determine the mobility of linkages. This criteria gives a lower bound on the number of degrees of freedom a mechanism has based solely on the number of links, the number of higher pairs and the number of lower pairs. For a planar mechanism, the degree of mobility is:

$$M = 3(n - 1) - J_h - 2J_l$$

where n is the number of links, J_h is the number of higher pairs and J_l is the number of lower pairs. M is a lower bound since, depending on the geometry of the objects, some links can be redundant and have no effect on constraining the degrees of freedom of a mechanism. Thus, the Kutzbach criteria is best used as a heuristic to determine the degrees of freedom of a mechanism. We are currently exploring this and other methods of analyzing such mechanisms.

4. References

- [1] "A Qualitative Physics based on Confluences", J. de Kleer and J. S. Brown, in *Qualitative Reasoning about Physical Systems*, D. Bobrow editor, MIT Press 1985.
- [2] "Qualitative Process Theory", K. Forbus, in *Qualitative Reasoning about Physical Systems*, D. Bobrow editor, MIT Press. 1985.
- [3] *The Kinematics of Machinery: Outline of a Theory of Machines*, Franz Reuleaux, 1876, Reprinted by Dover Publications Inc, 1963.
- [4] "Spatial Planning: A Configuration Space Approach", T. Lozano-Perez, *IEEE Transactions on Computers*, Vol. C-32, Number 2, pp 108-120.
- [5] *Theory of Machines and Mechanism*, J. E. Shigley and J. Uicker, McGraw-Hill Inc, 1980.