QUALITATIVE KINEMATICS IN MECHANISMS

Boi Faltings
Qualitative Reasoning Group
Department of Computer Science
University of Illinois
1304 W. Springfield Ave.
Urbana, Il. 61801
USA

Abstract

This paper investigates the problem of reasoning about the kinematic interactions between parts of a mechanism We introduce the concept of *Place Vocabularies* as a useful symbolic description of the possible interactions We examine the requirements for the representation and introduce a definition of place vocabularies that satisfies them We show how this representation can be computed from metric data and used as a basis for qualitative envisionments of mechanism behavior, and describe implemented algorithms to solve this problem

1 Introduction

The question of how to describe the physical world in a discrete, symbolic manner is a fundamental scientific issue in Artificial Intelligence As computers are increasingly applied to solve problems involving the physical world, it is also of great practical importance This paper addresses the problem of reasoning about the mechanical interactions of objects, which we call *qualitative kinematics* The goal is to develop methods for constructing a qualitative description of freedom of motion sufficient to understand a mechanism This work lies in the framework of Qualitative Physics ((FORBUS84|. [DEKLEER84]. [FORBUS81]. |DEK LEER79|. [HAYES79). [DEKLEER75]) The representation we construct can be used to compute an *envisionment* of the possible behaviors of a mechanism

A mechanical device achieves its function by the relatively constrained motion of its parts, a single rigid object considered by itself has no mechanical function The constraints on the motion of a part are determined by the points where it is in contact with other objects An adequate repre sentation of the function of a mechanism must therefore be based on the *connectivity* of the parts rather than the parts by themselves ([FFN87])

The connectivity of the parts varies with motion of the objects Changes in connectivity therefore define a tesselation of the space of possible positions of the objects into regions where the connectivity is constant The set of all such regions and their adjacency can serve to describe the possible behavior of the mechanism This is a generalization of the idea of a *place vocabulary* first proposed by Ken Forbus in ([FORBUS81|. [FOR BUS80]) In this paper, we show how the original concept, which was defined for the motion of points can be generalized to the case of moving geometric objects

An essential prerequisite to manipulate information about the physical world is an adequate representation of the continuum. There are two commonly used ways to do this (i) numerical solutions and (ii) reasoning in a fixed discretization A numerical solution exploits the mapping of all parameters of the system into the same domain (the number system) to achieve greater efficiency, but this fact also limits its power. For example, it is very hard to assess the effect of changes in parameters other than by explicit simulation for specific values, making problem solving very difficult Furthermore, because it is based on evaluating the description of the system at *points* in parameter space, the technique is necessarily incomplete: the point where undesired behavior occurs may not be among those evaluated. There has to be an explicit choice of parameter values for the evaluation, which makes it very difficult to reason with incomplete or uncertain information Much work has gone into this problem, the most successful results being Fuzzy Logic ([ZADEH79]) and the Shafer-Dempster theory ([SHAFER76]) but the solutions are still unsatisfactory An application of such quantitative methods to geometric problems has been studied in the ACRONYM system ([BROOKS81])

As an example of the second technique one might use a finite set of symbols to describe the standardized types and sizes of nuts and bolts and rules that state how these can fit together Such a technique is often very useful for practical applications, in fact, nuts and bolts *are* catalogued by such a fixed standardized scheme However, the scheme is missing *generativity* it can not deal with objects that fall outside its restricted symbol set Examples of this type of approach in AI applications include [STANFILL83) and [GELSEY87]

In the paradigm of qualitative reasoning, parameters are discretized in a *problem-dependent* manner A parameter is characterized by its relative magnitude with respect to a set of *landmark values,* which are parameter values where the solution of the problem considered changes This tech nique is both generative and complete, thus avoiding the above problems In this paper we show how to apply this paradigm to kinematics

E Davis ([DAVIS86]) has developed a set of special axioms for spatial reasoning in a naive physics framework Beyond the examples he has re searched it is not clear yet what the scope of application of this technique is As his approach uses less input information that ours, his reasoning does not have the level of detail that our approach gives This may be an advantage in many cases, but it seems that for mechanism analysis it is necessary to describe the interactions in greater detail

While the concepts we are proposing *are* very general, in the present research we restrict their application to *two-dimensional* analyses of *mechanisms* A mechanism can be defined as a *kinematic chain* used in such a way as to transmit or transform forces and motion A kinematic chain is a chain of *kinematic pairs,* which are pairs of elements linked together such that their relative motion is completely constrained By completely constrained" we mean that a motion of one of the elements completely determines the way that the other element must move As a further restriction on a kinematic chain, each element forms part of at most 2 kinematic pairs While this does not mean that each element only interacts with at most 2 others it assures that the function of the mechanism does not depend on the simultaneous interaction with multiple objects, so that each can be analyzed independently

We chose the mechanism domain for the following reasons

- Mechanisms exhibit interesting kinematics and there is practical interest in their analysis
- The freedom of the parts in a mechanism is very restricted, so that there are few parameters to be considered This makes the problem more tractable
- Mechanisms have been studied for a long time There exist many analyses which may help us evaluate our techniques
- Mechanisms are man-made devices Their kinematics are likely to be such that people are very good at analyzing them Discovery of techniques suitable to mechanisms might shed light on the nature of human spatial reasoning abilities.

We impose the two-dimensional limitation to simplify our algorithms and reduce the time spent on side issues, such as developing three-dimensional displays to debug our algorithms. However, we believe that almost all of the interresting phenomena can still be explored in two dimensions Furthermore, we plan to explore how our techniques might be extended to three dimensions as the project progresses further

In the next section, we describe in detail the concepts we are using and show how they apply to the problems we have described

2 Place Vocabularies

The concept of a place vocabulary originated in FROB ([FORBUS81], [FORBUS80)). This work investigated the qualitative analysis of the motion of point masses in polygonal, two-dimensional regions under the influence of gravity The qualitative representation of the geometry of the regions was derived by dividing the space into regions called *places,* and arranging them in a connectivity graph Because the work considered the motion of *points* the places could be obtained by a simple division of the physical space using horizontal and vertical lines In this paper we generalize the notion of a place to moving objects of finite dimensions by making a further abstraction from physical space to *configuration space* and then defining a useful way to break up this space into places

The position of a physical object can be described by a small set of parameters In the case of unrestricted motion in 3 dimensions, there are 3 Euclidian position parameters and 3 orientation parameters which completely determine the placement of the object In mechanisms, the motions of the parts are usually restricted by joints In most cases a single parameter suffices to describe its position We call the space spanned by the parameters characterizing the positions of all the objects *of* a mechanism its *configuration space* ([TLP79|. [TLP83]. [DON84]) At any time, the position of all the parts of the mechanism corresponds to a particular point in this space, we call this a *configuration.* As the parts of a mechanism mutually constrain their positions, the configuration space consists of regions corresponding to legal and illegal configurations. We call the union of all legal regions the *free space* and its complement the *blocked space* The boundaries between the 2 regions are formed by *configuration space constraints* They are defined by pairs of objects and correspond to either a vertex or a boundary segment of one object touching the boundary of the other We call the former *vertex constraints* and the latter *boundary constraints* From any configuration on the boundary of free space, points in both blocked in free space can be reached by arbitrarily small motions This is the case only if the 2 objects touch Boundary and vertex constraints cover all possible cases of this and thus are the only possible boundaries between free and blocked space Each constraint is applicable only within a certain interval, this can be expressed by *applicability constraints* ([DON84]) Applicability constraints restrict the domain of configuration space within which the associated constraint is valid The actual boundaries between regions of free and blocked space are defined by the envelope of the constraints Constraints are *subsumed* (and not applicable) wherever they fall inside this envelope

In general, we can think of a mechanism as having an underlying configuration space of a finite, but possibly enormous, number of dimensions In the analysis, one only considers interactions between small numbers of parts, which corresponds to working in subspaces of the full C-space

To arrive at a useful definition for places in configuration space, we consider the following 3 requirements that the place vocabulary must satisfy to capture the kinematics of mechanism:

1. The places must distinguish at which points the objects are in contact.
2. For each contact point, it must be possible to give the contact force in an arbitrary qualitative coordinate system, and
3. For motions specified in an arbitrary qualitative coordinate system, it must be possible to enumerate the set of possible place transitions.

These conditions ensure that the place vocabulary can serve as a basis for qualitative spatial reasoning about motion The place vocabulary provides a symbolic framework for spatial reasoning by providing answers to the following questions;

- Given a set of objects, what configurations of contact are possible?
- Given that some objects are moving in a particular direction, what changes *of* contact can result?

This information, combined with qualitative coordinate systems and dynamical information (i.e.. information about forces, expressed using Qualitative Process Theory [FORBUS84]) provides the information required to reason about motion

How do we choose a division of configuration space such that the resulting regions are places satisfying these requirements? Requirement (1) means that there *are* several classes of places full-dimensional places where no contact points between objects exist and they are free to move in any direction, and lower-dimensional places where the objects *are* in contact and some degrees of freedom are thus eliminated. Note that the boundaries of a place of dimension d are formed by places of dimension d-1 This type of arrangement is called a *cell complex in* topology. Note also that all places of dimensions less than that of the configuration space are defined by segments of the constraints or intersections of these

If the boundaries of the places are described in algebraic form it is possible to satisfy requirements (2) and (3) by computing the directions of the normal forces and boundaries. If such a description is not available, this information may be found by computing the desired quantities by local analysis for a sample point In order to give unique results for these quantities in a given qualitative coordinate system, the places have to be *monotonous* in the coordinate system, i.e. the qualitative directions that are computed may not change. Additional divisions into monotonous segments have to be made where this is not the case

As defined so far. the places only distinguish connected regions of free space This is a very weak description, as the number of different constraint segments bounding such a region may be very large This in turn may result in an impractically large number of ambiguities in the analysis of the mechanism. We therefore break up these regions further into *quasi-convex* cells. By this term, we mean cells defined by a set of bounding curves (surfaces) C,(x) = 0 such that all of the points in the cell satisfy a certain conjunction of inequalities on the signs of the C, In order to ensure this property, we sometimes need additional curves to further tesselate places, these are called *free-space divisions* and defined using the algebraic curves of the applicability constraints While this implicitly defines a further tesselation. it is introduced purely to compensate for idiosyncracies of the algebraic representation, therefore we do not include it in the place vocabulary definition Details of how regions are broken up can be found in ([FAL87]).

The places where objects are in contact are given by the roots of one or more of the constraint polynomials They can be expressed in the same formalism by replacing the appropriate inequalities by equalities We have thus obtained a uniform definition of places for any number of degrees of freedom

The places thus constructed are arranged in the *place graph* The vertices of this graph are the places, and each places is connected by edges to the places forming its boundaries, and possibly to adjacent places within the same connected region. This place graph forms the spatial substrate for an *envisionment,* which is a directed graph describing the possible qualitative states (places) and transitions between them ([FORBUS84]. (DEKLEER84). [FORBUS81], [DEKLEER79]. (DEKLEER75)]

2.1 Example of a Place Vocabulary

In this section, we give a brief example of what the place vocabulary for an actual mechanism looks like and indicate how it can be used to reason about the function of a mechanism. Consider the escapement shown in Fig. 1 below. Given that the freedom of the parts has been identified, the configuration space for this interaction is 2 dimensional, spanned by the parameters o for the orientation of the wheel and w for that of the anchor The valid configuration space constraints for this example are shown as solid lines in figure 2 below. The program in this case finds 4 faces of the graph formed by constraint intersections These form the boundaries

[1] we call this quasi-convex because if the boundaries of the cell are straight lines it is just the convexity property

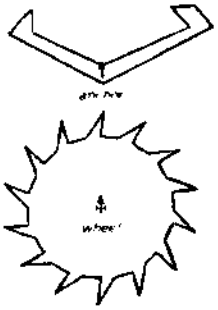

Figure 1: Escapement Example

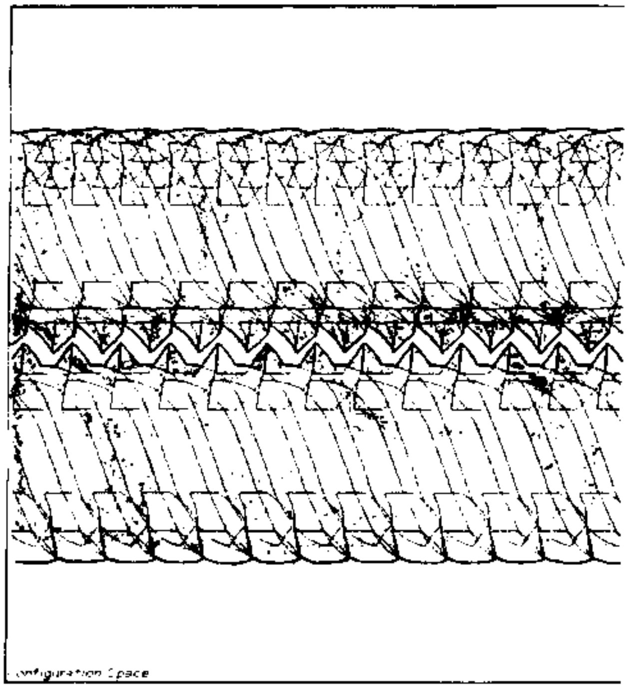

Figure 2: The configuration space for the escapement example. Note that the space is a torus surface and thus wraps around on the sides. The horizontal direction is ϕ, the vertical ψ.

of 2 free-space regions one corresponding to the normal operation of the escapement and one corresponding to the anchor turned over examples of these are shown in figure 3 below We discuss the region corresponding to the normal operation of the escapement Figure 4 shows an enlarged section of this part of the configuration space The program adds divisions of the free space which are indicated by the dashed lines The region is bounded by sequences of constraint segments corresponding to the right side and the left side of the anchor touching the wheel These form the 1-dimensional places the set of configurations where there exists a point of contact between a certain vertex and a certain boundary segment of the object satisfies the corresponding constraint curve For each tooth, the program finds 1-dimensional places corresponding to the 6 different configurations of touch shown in figure 5 below The constraint segments corresponding to these are indicated by letters in figure 4 In cases A). C) and D). there exists a point where the qualitative relation between the configuration space parameters changes they *are* thus further subdivided by the program Note that in the actual place graph, these places *art* further broken up by intersecting free space divisions The places shown in each row of figure 5 are connected in a sequence as indicated by the arrows. There are 14 teeth on the wheel so there are 14 periodic repetitions of the places shown In the actual operation of the escapement

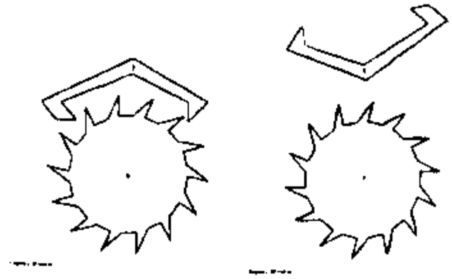

Figure 3. Configurations in the *1* legal connected regions

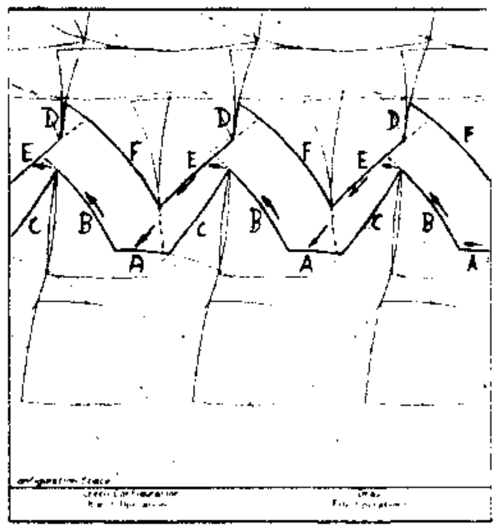

Figure 4 An enlarged section of the configuration space. The letters refer to the configurations shown in figure of the arrows indicate motion of escapement during normal operation

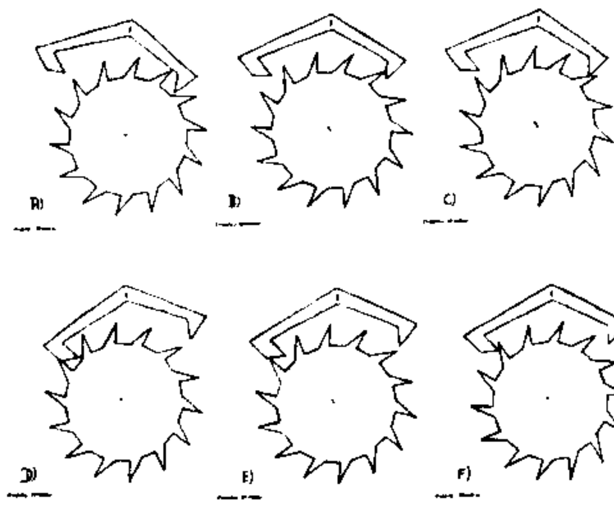

Figure 5: Sample configurations for the 1-dimensional places in the escapement example.

the wheel is moving clockwise and the anchor alternatively touches the wheel with its right and its left end This motion is indicated by the arrows in figure 4. The sequence of places it passes through is then A) → B) → intermittend motion → E) —> intermittend moti → A)" .... where A)' refers to the next periodic repetition of A). The intermittend motions, where no contact between the objects exists, are represented by 2-dimensional places

The intersection points between the one-dimensional places are the 0-dimensionai places of the place vocabulary The free space region between the 2 sequences is broken up by several free-space divisions in order to satisfy the quasi-convexity criterion. These are indicated by the dashed lines in figure 4. Note that each free space division begins at a vertex in configuration space, as the defining applicability constraints intersect the associated constraints at their endpoints The tesselation introduced by the these divisions defines the set of 2-dimensional places Note that the tesselation strongly reduces the number of adjacent 1-dimensional places and thus serves to reduce ambiguities in the envisionment that can be produced using the place graph In this example, the program generates 3 2-dimensional places for each tooth

We can express the adjacencies between the places in the adjacency graph, part of which is shown in Fig 6 The adjacency relation defining the edges in the graph is the boundary relation between the places A sequence of motions of the mechanism can be described as a sequence of place transitions in the place graph: and any sequence of places admitted by the adjacency relation is an achievable motion In the place graph free space divisions are represented by explicit objects In figure 6. the links between identical free-space divisions have been eliminated to allow a planar display of the graph

We outline how an envisionment of the escapement s operation can be obtained using the place graph For each place, the set of possible transitions is given as the set of adjacent places We describe the velocities and forces or moments of the objects by their signs Note that the 2 quantities are related by a simple qualitative differential equation Each of the one-dimensional places enforces a monotone relation between the configuration space parameters This results in a constant relation between the qualitative parameters for the motion and forces within these places The 0-dimensional places are contained in 1-dimensional places and so share these characteristics For the 2-dimensional places, there is no contact between the objects and they therefore do not influence each other. Details of the actual qualitative simulation are described in ([IFFN87])

As a final point, note that we have not described the other region of free space This region is also broken up into subregions by free-space divisions and forms a disjoint component of the place graph.

3 Descriptive Power of the Place Vocabulary

The descriptive power of the place vocabulary as a knowledge representation is characterized by the following three properties First, it makes the necessary geometric distinctions to support qualitative mechanical analysis. Second, it fully describes the topology of configuration space The existence of a legal trajectory between 2 points in C-space is represented by a path in the place graph, and all topologically distinct trajectories are distinguished in the place graph Third, there exists a simple mapping from the symbolic description back to the original domain Trajectories in configuration space corresponding to paths in the place graph can be assembled from trajectories through the regions corresponding to the places The place vocabulary allows us to continuously maintain a set of applicable constraints on the configuration, so that such trajectories can be found by application of relaxation algorithms such as those described in ([KHATIB85]). The existence of such a mapping is important for any knowledge representation because it defines the meaning of its instances

Note that every non-convex intersection of constraints forces a change in the qualitative description of the constraint surfaces involved There fore, the distinctions the place vocabulary makes are also necessary, and thus the place graph is the simplest symbolic description that satisfies our requirements

4 Computation of Place Vocabularies

In this section, we give a brief description of the algorithm to compute place vocabularies from a metric diagram for the case of a kinematic pair of 2 objects in 2 dimensions, with 1 degree of freedom each With very few exceptions, this case covers all interactions in 2-dimensional mechanisms, such as gearwheels, ratchets escapements and so forth. We have implemented a complete system to handle this case for objects whose boundary curves are either straight lines or arcs As the kinematic

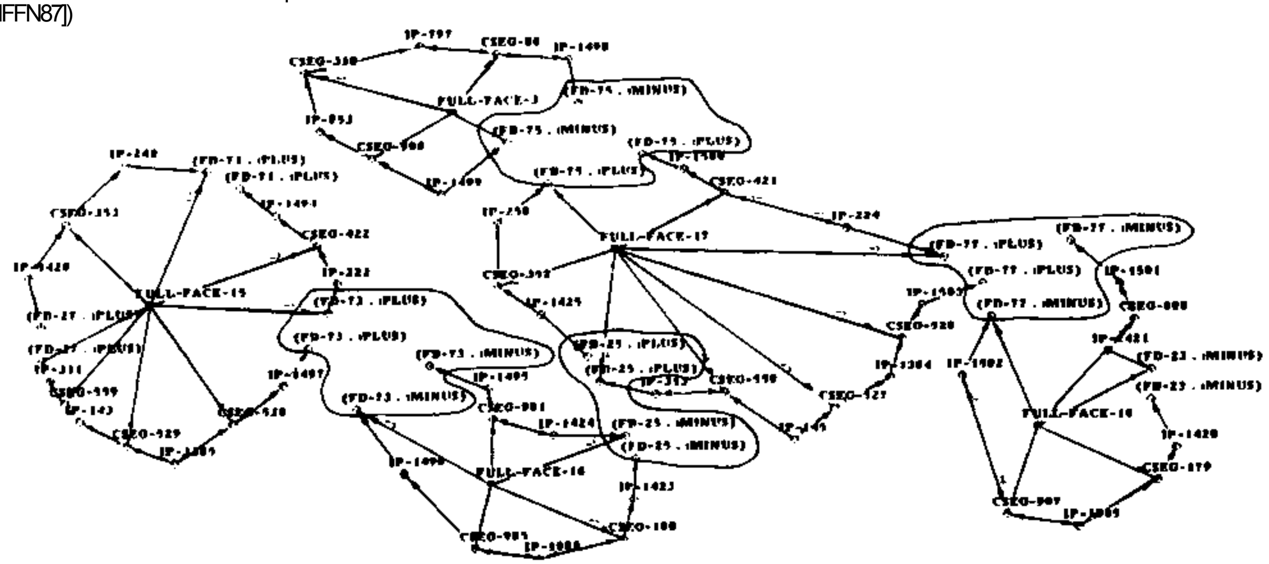

Figure 6: Part of the Place Graph for the escapement. 2-dimensional places are labelled with FULL-FACE. 1-dimensional ones with CSEG, and 0-dimensional ones with IP. Free-space divisions are indicated by (FD-n.:PLUS) and (FD-n.:MINUS) to indicate the 2 sides.

pairs in the mechanism form a kinematic chain a description of it can be built up from descriptions of the underlying kinematic pairs

The algorithm we are describing is *not* designed to analyze joint configurations This problem has been excluded from this research because

- There already exist concise theories of joints and their composition
- The space of possible joints is finite, eliminating the need for a generative system

Thus, we assume that the freedom of each part is explicitly stated as part of the input information We distinguish the 2 cases of translational and rotational freedom As a symbolic description of the parts themselves, we assume the following boundary-based representation

- an *object* consists of a set of *boundaries* and a local coordinate system centered at the *reference point*
- a *boundary* is an alternating sequence of *vertices* and *boundary segments* such that each boundary segment connects the 2 vertices adjacent to it and the first and last vertex of the list are identical
- a *vertex* corresponds to a discontinuity in the direction of the boundary, or a change in curvature or the algebraic form

In our implementation we allow the algebraic type of the boundary curves to be either straight lines or arcs Allowing general algebraic curves would require a general algebraic engine to solve problems like determining intersections and extrema while the algorithms for such systems exist it is not the focus of our research to implement them Note that the sym bolic representation we assume could be readily obtained by rearranging the output representation of a vision system such as Smoothed Local Symmetries ([BRADY86])

4.1 Computing Place Vocabularies for Kinematic Pairs

The algorithm first computes the set of all possible constraints that the 2 objects may form For vertex constraints which correspond to a vertex of one object touching a boundary segment of the other, the constraint is given by the condition that the vertex must lie on the boundary segment Boundary constraints in our implementation are formed by arcs They can be handled in the same *manner* by observing that an arc touches a straight line exactly when its center lies on an imaginary line offset by the radius of the circle from the actual one (see in figure *1* In the case of rotational freedom for each possible configuration where the objects touch there exists a dual configuration where the same contact exists In this case, there then exist 2 copies of each constraint to account for the 2 cases For each constraint the program computes

- its equation *and* tangents as function of the configuration space parameters
- a parameterization in terms of a parameter corresponding to the location of the point of touch on the boundary segment
- the set of endpoints of the constraint, given by the condition'that the contact occurs at the ends of the boundary segment
- a set of applicability constraints which separate the valid segment of the constraint curve between the endpoints from the rest

As the boundary segments are connected at their endpoints each endpoint is common to 2 boundary segments, and the configurations where 2 such endpoints touch each other are common endpoints of up to 4 constraints We call such configurations *touchpoints*

Depending on the relative arrangements of the objects only a certain subset of these constraints and touchpoints will actually be valid For example, for 2 rotationally attached objects a touchpoint will only exist when the distance between the 2 centers is in the interval between the sum and the difference of the radii of the 2 vertices that generate the touchpoint This set of actually valid elements is determined in the second stage of the computation

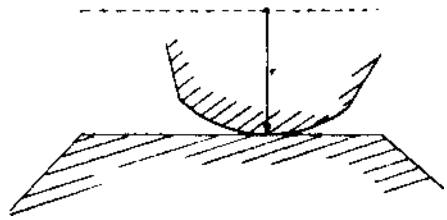

Figure 7: The arc touches the lines exactly when its center lies on one of the dashed lines

The constraints may intersect each other not only at touchpoints but can also form *subsumptxon* intersections For example, in the example of 2 gearwheels, there exists a point where the contact at one tooth is subsumed by the next Such intersections can occur anywhere along a constraint segment The 3rd stage of the algorithm thus tests ail pairs of constraints for intersection This is implemented by testing endpoints of constraint segments As the constraints are 2nd degree curves, there is a possibility that 2 constraint segments intersect in 2 points, which would not be detected by this test. This case can be dealt with using algebraic decision techniques based on Sturm's theorem So far. we have not implemented this as we have not observed this case in an actual mechanism

Finally, the boundaries of the connected regions of free space can be found by tracing the faces of the graph formed by the constraints and their intersections Only a subset of these qualifies as actual free-space regions These are characterized by the condition that the inside of the face lies on the legal side of all its bounding constraint segments It is still possible that a face found in this way is not part of free space This can be determined by testing a sample point for the existence of overlap between the objects

In the case of rotational attachment, the topological form of the configuration space may be a cylinder or torus On such surfaces there may exist regions that *are* bounded by more 2 distinct connected boundaries. This corresponds to having 2 faces of the graph forming the boundary of a single region The next step in the algorithm is thus the association of the faces to find such pairs It is also possible that a region of free-space has several disjoint boundaries this case is tested for in a final stage

Finally we introduce further tesselations into the regions to achieve the quasi-convexity property For each region, the applicability constraints for the constraints bounding the region are tested for valid segments within the region Each such segment forms a free space division of the region This step is carried out separately from the constraint intersection tests because the association with regions significantly reduces the complexity

4.2 Composition of Place Vocabularies

Place composition is the intersection of several different place vocabularies sharing one or more configuration space *parameters* to find a single description of the places allowed by the objects in conjunction There are 2 different cases in which such place composition is necessary to compose place vocabularies generated by different boundaries of the same objects, and for composing place vocabularies for kinematic pairs into descriptions of kinematic chains We refer to the first case as *codxmensxonat* composition, and to the second case as *chain* composition

4.2.1 Co-dimensional Place Composition

To compose places in the same configuration space, we have to find the regions formed by intersections of the regions forming the places In the general case, this is not an easy problem (see [PS85]) In the case of 2-dimensional configuration space, the boundaries of the places are one-dimensional and have points as intersections. We can find the intersections between the regions by finding all intersections and tracing out the faces of the resulting graphs, where again an association step is needed to handle regions with multiple boundaries Note that because the intersections of quasi-convex regions are again quasi-convex the resulting regions will be legal places This process is the same as that used for combining the tesselations given by concave chains

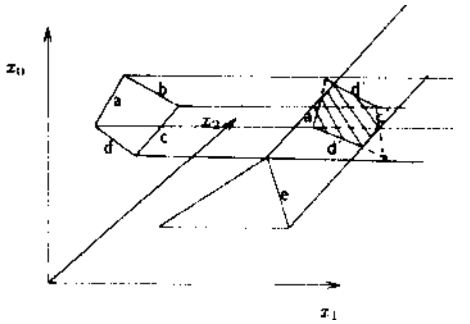

Figure 8: The intersections of the backprojections of a, b, c and d define a face bounding the intersection volume on the backprojection of e.

4.2.2 Chain Composition

In the problem of chain composition we have 2 place vocabularies whose configuration spaces have one or more parameters in common The regions corresponding to the places must be ' backprojected" into the product space of the 2 C spaces The composed place vocabulary is defined by the regions formed by the intersections of these backprojections Like the co-dimensional problem, in the general case, this is also a very difficult problem

In the case of 2-dimensional configuration spaces, however, the place vocabularies share at most one single parameter The intersection problem is then very simple as it is reduced to interval intersection tests in that parameter This is because the plares formed by the backprojections of the constraint segments bounding the places intersect if and only if the intervals they cover in the common parameter intersect Furthermore, none of these intersections can be subsumed by other backprojections. so that all the intersections of the backprojections actually occur as boundaries of the intersecting volume The composition process is illustrated in Fig 8 below

4.3 Implementation

We have implemented the algorithm for computing place vocabularies for kinematic pairs in Common Lisp on a Symbolics Lisp machine The initial instantiation of the constraints is rather slow due to the symbolic algebra required (for the escapement example this stage takes about SO minutes) The further computation of the place vocabulary for the escapement takes about 35 minutes, with completely unoptimized code Because it is intended to be used for further research the implementation uses a symbolic algebra system for manipulation of the information about the objects The computation could be sped up markedly by computing these directly Also, the algorithms could be very well implemented in parallel The program has also run on other examples like a ratchet and gearwheels, and we are currently in the process of analyzing the interactions in an actual clockwork, the QRG clock shown in figure 8

5 Conclusions

We have presented an application of the qualitative reasoning approach to mechanism kinematics We have introduced the concept of place vocabularies and shown that it is a well-defined and useful representation of mechanism kinematics We have presented an implemented algorithm to compute place vocabularies and given an example of its application.

We have not investigated as yet how the techniques we have implemented can be generalized to 3 dimensions or more degrees of freedom

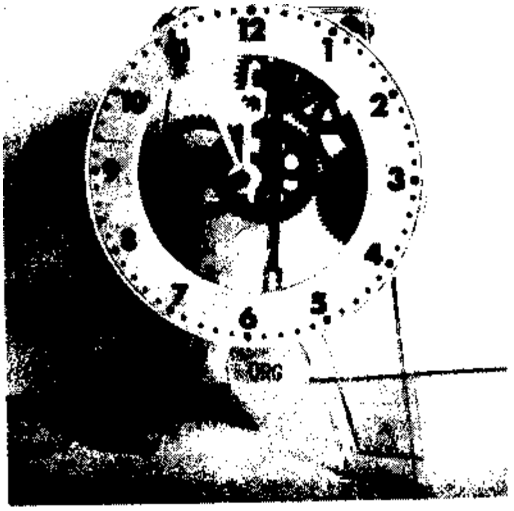

Figure 8: The QRG clock

There exist algebraic techniques based on decision methods ((TAR48]. [SCHS83). [BKR85] [KY85]) which provide an existance proof of such algorithms but they are rather opaque and inefficient Because the input and output of the algebraic algorithms are the same as those of a specialized algorithm, it may be possible to specialize them to obtain algorithms similar to the ones we have presented for more general cases This might be a promising research strategy for extending the range of validity of the current implementation

Currently, we are investigating ways to compute the place vocabulary in a qualitative manner with partial information Such algorithms will allow us to apply the place vocabulary concept to mechanism design problems

A question we are unfortunately still far from answering is that of the human spatial ability The problem here is that there is no well defined class of problems that people can solve perfectly, which prohibits formulation of simple requirements for a theory. By showing more clearly what the actual requirements for *perfect* reasoning are, we hope to have provided a basis for deriving cognitive theories by weakening the techniques

5.1 Acknowledgements

I would like to thank Ken Forbus for much helpful advice and comment IBM for partial support through an IBM graduate fellowship, and ONR for support through contract no N00014-85-K-0225.

References

[BKR85| Ben-Or, D. Kozen, J. Reif "The Complexity of Elementary Algebra and Geometry". Journal of Computer and System Sciences 32. 1986. pp 251-264

[BRADY86] H. Asada, M. Brady "The Curvature Primal Sketch". IEEE Transactions on Pattern Analysis and Machine Intelligence. 8(1986). pp. 2-14

[BROOKS81) Rodney A. Brooks Symbolic Reasoning Among 3-D Models and 2-D Images. Artificial Intelligence 17(1981). pp 28S-348

[DAVIS36] E. Davis A Logical Framework for Solid Object Physics * New York University Technical Report 245. October 1986

[DEKLEER75] J. DeKleer "Qualitative and Quantitative Knowledge in Classical Mechanics MIT AI TR-352 1975

[DEKLEER79] J. DeKleer "Causal and Teleological Reasoning in Circuit Recognition MIT AI Lab TR-529 September 1979

(DEKLEER84] J. DeKleer, D. Bobrow Qualitative Reasoning with Higher-Order Derivatives Proceedings of the National Conference on AI. Austin August 1984

|DON84] Bruce Donald Motion Planning with Six Degrees of Freedom MIT AI Tech Report 791. May 1984

(FAL87| Boi Faltings Algorithms for Qualitative Kinematics in Mechanisms". University of Illinois Tech Report. May 1987

[FFN87| Ken Forbus, Paul Nielsen, Boi Faltings The Inferential Structure of Qualitative Kinematics". Proceedings of IJCAI 1987

[FORBUS80) Ken Forbus Spatial and Qualitative Aspects of Reasoning about Motion . Proceedings of the National Conference on AI Stanford. August 1980

[FORBUS81] Ken Forbus A Study of Qualitative and Geometric Knowledge in Reasoning about Motion'. MIT AI TR 615. February 1981

[FORBUS84] Ken Forbus Qualitative Process Theory MIT AI TR 789. July 1984

(GELSEY87| Andrew Gelsey Automated Reasoning about Machine Geometry and Kinematics . 3rd IEEE Conference on AI Applications Orlando. Fla . Feb 1987

[HAYES79] B. Hayes The Naive Physics Manifesto . in Expert Systems in the Micro-Electronic Age D Michie (ed) Edinburgh University Press. May 1979

[KHAT85) O. Khatib "Real-Time Obstacle Avoidance for Manipulators and Mobile Robots". Proceedings 2nd IEEE International Conf on Robotics and Automation. St Louis. March 1985

(KY85) D. Kozen, C. K. Yap: Algebraic Cell Decomposition in NC". extended abstract. Proceedings of the 26th FOCS. Oct 1985

(SCHS83| J. Schwartz, M. Sharir On the Piano Movers Problem II General Techniques for Computing Topological Properties of Real Algebraic Manifolds" Advances in Applied Mathematics 4 (1983). pp 298-351

[SHAFER76| G. Shafer: A Mathematical Theory of Uncertainty' . Princeton University Press. 1976

[STAN83] Craig Stanfill The Decomposition of a large Domain: Reasoning about Machines". Proceedings of the National Conference on AI. August 1983

[TAR48] Alfred Tarski "A Decision Method for Elementary Algebra and Geometry". University of California Press. 1948

[TLP79] T. Lozano-Peres, M. Wesley "An Algorithm for Planning Collision-Free Paths Among Polyhedral Obstacles". Comm. of the ACM. Vol. 22 (1979). pp. 560-570

[TLP83] R. Brooks. T. Losano-Perez A Subdivision Algorithm in Configuration Space For Findpath With Rotation'. Proceedings of the IJCAI 1983. pp 799-806

[ZADEH79) L.A. Zadeh: "A Theory of Approximate Reasoning", in: Machine Intelligence 9. J Hayes. D Michie. L.I. Mikulich (Eds). John Wiley and Sons, New York 1979