Extracting Articulation Models from CAD Models of Parts with Curved Surfaces

Rajarishi Sinha[1,*], Satyandra K. Gupta[2], Christiaan J.J. Paredis[1], Pradeep K. Khosla[1]

---

[1] Institute for Complex Engineered Systems, Carnegie Mellon University, Pittsburgh, PA 15213.
[*] Corresponding author; rsinha@cs.cmu.edu
[2] Department of Mechanical Engineering, University of Maryland, College Park, MD 20742.

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT

Degrees of freedom in an assembly are realized by creating mating features that permit relative motion between parts. In complex assemblies, interactions between individual degrees of freedom may result in a behavior different from the intended behavior. In addition, current methods perform assembly reasoning by approximating curved surfaces as piecewise linear surfaces. Therefore, it is important to be able to: reason about assemblies using exact representations of curved surfaces; verify global motion behavior of parts in the assembly; and create motion simulations of the assembly by examination of the geometry. In this paper, we present a linear algebraic constraint method to automatically construct the space of allowed instantaneous motions of an assembly from the geometry of its constituent parts. Our work builds on previous work on linear contact mechanics and on our previous work on curved surface contact mechanics. We enumerate the conditions under which general curved surfaces can be represented using a finite number of constraints linear in the instantaneous velocities. We compose such constraints to build a space of allowed instantaneous velocities for the assembly. The space is then described as a set-theoretic sum of contact-preserving and contact-breaking motion sub-spaces. Analysis of each subspace provides feedback to the designer, which we demonstrate through the use of an example assembly – a 4-part arm. Finally, the results of the analysis of a 4-bar linkage are compared to those from mechanism theory.

1. INTRODUCTION

Assemblies are composed from parts. The geometry of the parts imposes certain restrictions on the way that they can be assembled, and also on the way that they move relative to one another. This in turn has a bearing on their static stability, kinematic behavior, and dynamic performance characteristics. However, none of this information exists initially. It must be derived from the CAD models of the individual parts, and from the relative positions of these parts in the final assembly. In order that the collection of parts can be grouped together to form the assembly representation of the artifact, joints between parts must be specified. The joints between parts are defined by the designer at the conceptual design level to meet certain functional requirements of the assembly. Thus, there can be two types of constraints between parts, namely, constraints induced by the geometry of parts, and constraints introduced by the designer to satisfy functional requirements of the assembly. Both these types of constraints interact to produce a resultant behavior of a joint.

Articulated devices are realized through contacts between parts. In order to verify that a complex device is a correct spatial realization of the intended functional design concept, we need to extract its behavior from its geometric representation. Techniques have been developed to predict the instantaneous degrees of freedom from the CAD models of parts composed of polygonal planar faces [Hirai and Asada, 1993; Mattikalli et al., 1994]. However, these techniques handle only parts with planar faces; most engineering devices have curved parts. When curved parts are approximated as piecewise planar parts, erroneous results are possible.

In this research, we present a methodology that extends our earlier work on contact surfaces. That work [Sinha et al., 1998] reasons about the degrees of freedom at each joint, based on surface mating constraints, that are in turn obtained from analyzing the nature of body to body contact. Non-penetration constraints are imposed along the boundary of each contact surface in the form of algebraic inequalities. It is shown that a finite number of non-penetration conditions are representative of the entire surface in contact. Using linear programming methods, instantaneous velocities and accelerations for each pair of bodies are computed. In this work, we obtain a set of properties that must be satisfied by a general contact surface in order to preserve the linearity of the model. We describe a method by which the space of allowable motions in the assembly can be described concisely.

Such a methodology is useful in that it can provide useful feedback to the designer. He or she can determine which components are free to move in the assembly. The procedure can be completely automated, so that there are no errors induced by user interaction. This eliminates the possibility of input errors. In addition, since the method is algebraic and uses linear programming, it is fast and is valid for all possible surface contacts, unlike rule-based systems that operate on a feature level. This method will also account for contact surfaces with incomplete geometry (such as portions of planes, cylinders, or spheres).

2. REVIEW OF PREVIOUS WORK

2.1. Early Work on Contact Mechanics

Previous work on planar contact mechanics and screw theory prepared the foundation for this work. Ohwovoriole and Roth [1981] showed that unidirectional constraints can be modeled as screws. Hirai and Asada [1993] described the allowable motions of a part using polyhedral convex cones to represent the space of movement. Mattikalli and Khosla [1991] described a method to obtain degrees of freedom from component mating constraints, wherein they use a unit sphere to represent the space of all available degrees of freedom.

Some issues have not yet been addressed satisfactorily in these frameworks. Current shortcomings in articulation research include:
1. Only bodies described (or approximated) by planar surfaces are considered.
2. Current techniques are local; global interaction (propagation of constraints beyond the point where they are induced) is not satisfactory.
3. Current simulation techniques do not detect incorrect/incomplete inputs; there is no verification for correctness of the articulation representation and for the compound effect of geometric interactions and physics-based interactions.

2.2. Contact Mechanics for Planar Contacts

A part in an assembly is in physical contact with one or more other parts. The nature of these contacts can provide useful information about the types and limits of the degrees of freedom at these contact points. Some of these contacts induce surface mating constraints, leading to the formation of a joint. Other contacts are *incidental*, in that they may introduce limits on the degrees of freedom of the joint [Rajan et al., 1997; Rajan and Nof, 1997]. Reasoning about these constraints provides the designer with valuable insight into the instantaneous degrees of freedom of the assembly.

Other researchers ([Mattikalli et al., 1994], [Baraff and Mattikalli, 1993]) have worked with polygonal bodies and polygonal surfaces of contact. They approximate curved planar boundaries using straight lines, and use linear programming techniques to solve the contact problem.

When a pair of parts are in contact with each other, it implies that there is no inter-penetration between the parts at the contact surfaces. This *non-penetration condition* at a point can be written as [Baraff and Mattikalli, 1993]:

$$(\vec{v} + \vec{w} \times \vec{r}) \bullet \vec{n} \geq 0 \qquad \text{Eq. 1}$$

where $\vec{v}$ is the relative translational velocity between two parts, $\vec{w}$ is the relative angular velocity between two parts, $\vec{r}$ is the position of the point and $\vec{n}$ is the normal at a point of contact on the surface of contact. This equation is linear in $\vec{v}$ and $\vec{w}$. We define the *generalized velocity vector* as $[\vec{v} \ \vec{w}]$. Equation 1 implies that the generalized velocity vector for relative motion between two

points, one on each contact surface, should not have a component opposite to the normal to the base surface. A component into the base surface will imply penetration.

To prevent the penetration of one part into the other, Equation 1 must be satisfied at every point on the surface of contact and on the boundary of the surface of contact. Such a method of expressing planar contact between two bodies has been used before. For example, see Baraff and Mattikalli [1993], where the authors use non-penetration conditions to determine the impending motion direction of polyhedral rigid bodies in contact.

A closed planar surface (or *patch*) is bounded by a finite set of curves; these curves may be straight line segments or curved line segments. Any point in the interior of the patch can be expressed as a linear combination of points at the vertices of the boundary of the convex hull of the patch. As long as the convex hull has a finite number of vertices, there will be a finite number of non-penetration conditions, all of which will be linear in $\vec{v}$ and $\vec{w}$.

When extending to curved surface contacts, it is desirable to preserve the linearity of the formulation for reasons of computational efficiency; linearity allows for easier search and boundary enumeration.

2.3. Extending Contact Mechanics to Curved Surfaces

In a previous paper [Sinha et al., 1998], we extended the results obtained for planar surface contacts by showing that similar results can also be obtained for spherical and cylindrical surfaces defined by edges which are great arcs (for spherical surfaces) or straight lines and circular arcs (for cylindrical surfaces).

Spherical contact surfaces in contact always result in unconstrained rotations, because they share a common center. As before, the non-penetration conditions must be written at the vertices of the convex cone for the given spherical contact surface. However, since the cost of computing the convex cone is high, we chose to generate the non-penetration condition at the vertices of the spherical patch. This will not influence the final result.

Non-penetration at every point in a contact patch on a cylindrical surface with boundary segments that are exclusively constant-z and constant-θ segments can be represented entirely by non-penetration at the vertices of these segments.

3. GENERALIZED CONTACT MECHANICS

The discussion in Section 2 illustrates that for certain curved surfaces, if the boundary can be described by a finite number of segments all possessing certain properties, then the non-penetration condition at any point on the curved surface is satisfied if the non-penetration condition is satisfied at the finite number of boundary vertices.

The nature of the physical contact between a pair of parts in an assembly provides useful information about the types and limits of the degrees of freedom. Some of these contacts induce surface mating constraints, leading to the formation of a joint. Other contacts are *incidental*, in that they introduce limits on the degrees of freedom of the joint. When two parts are in contact with each other, it implies that there is no inter-penetration between the parts at every point on the contact surfaces. Non-penetration conditions can be written as linear inequalities in the instantaneous velocity, which, when taken together, describe a linear subspace.

The following Proposition will be used to establish a theoretical basis for the linear treatment of curved surfaces in assembly modeling.

Proposition 1. Given:

1. *A continuous curve $C(\lambda): \lambda \in [0,1] \rightarrow R^3$.*
2. *$C(0)=P_1$ and $C(1)=P_2$; $P_1,P_2 \in R^3$.*
3. *C lies on a parametrizable, differentiable contact surface S formed between two bodies A and B.*

Then non-penetration (by Equation 1) at P_1 and P_2 implies non-penetration at any point on C, if and only if:

1. *C is a circular arc, possibly with infinite radius (limiting case of a straight line)*
2. *The unit normal to C at any point along C is equal to the unit normal to S at that point.*

Proof. We prove Proposition 1 by showing that given Equation 1 written at P_1 and P_2, Equation 1 holds along a set of points between P_1 and P_2.

Equation 1 written at P_1 is:

$$(\vec{v} + \vec{w} \times \vec{r_1}) \bullet \vec{n_1} \geq 0 \qquad\qquad \textbf{Eq. 2}$$

and at P_2 is:

$$(\vec{v} + \vec{w} \times \vec{r_2}) \bullet \vec{n_2} \geq 0 \qquad\qquad \textbf{Eq. 3}$$

4

where $\vec{v}$ is the relative translational velocity between the bodies A and B, $\vec{w}$ is the relative angular velocity between the two bodies, $\vec{r}_1$ and $\vec{r}_2$ are the position vectors of P$_1$ and P$_2$, respectively. $\vec{n}_1$ and $\vec{n}_2$ are the normals to S at P$_1$ and P$_2$ respectively. Forming a linear combination of Equation 2 and Equation 3, we get:

$$\boldsymbol{l}(\vec{v} + \vec{w} \times \vec{r}_1) \bullet \vec{n}_1 + (1 - \boldsymbol{l})(\vec{v} + \vec{w} \times \vec{r}_2) \bullet \vec{n}_2 \geq 0 \text{ with } \boldsymbol{l} \in [0,1] \qquad \textbf{Eq. 4}$$

Rearranging terms in Equation 4, we get:

$$(\boldsymbol{l}\vec{n}_1 + (1 - \boldsymbol{l})\vec{n}_2) \bullet \vec{v} + \boldsymbol{l}\vec{n}_1 \bullet \vec{w} \times \vec{r}_1 + (1 - \boldsymbol{l})\vec{n}_2 \bullet \vec{w} \times \vec{r}_2 \geq 0 \qquad \textbf{Eq. 5}$$

For Equation 5 to be true and of the form of Equation 1, the following would have to be true $\forall \vec{v}, \vec{w}$:

$$\vec{n} = \boldsymbol{l}\vec{n}_1 + (1 - \boldsymbol{l})\vec{n}_2 \qquad \textbf{Eq. 6}$$

and

$$\boldsymbol{l}\vec{n}_1 \bullet \vec{w} \times \vec{r}_1 + (1 - \boldsymbol{l})\vec{n}_2 \bullet \vec{w} \times \vec{r}_2 = \vec{n} \bullet \vec{w} \times \vec{r} \qquad \textbf{Eq. 7}$$

Equation 6 is an expression which indicates that $\vec{n}$ spans all the normals from $\vec{n}_1$ and $\vec{n}_2$. For Equation 7 to be satisfied for all $\vec{w}$, it is sufficient to show that:

$$\boldsymbol{l}\vec{n}_1 \times \vec{r}_1 + (1 - \boldsymbol{l})\vec{n}_2 \times \vec{r}_2 = \{\boldsymbol{l}\vec{n}_1 + (1 - \boldsymbol{l})\vec{n}_2\} \times \vec{r} \qquad \textbf{Eq. 8}$$

Equation 8 is an expression for the generatrix or *trace* [Gray, 1998] $\vec{r}$ of C which generates a locus of points where Equation 1 is satisfied, given that it is satisfied at P$_1$ and P$_2$. Taking the dot product of Equation 8 with $\vec{n}_1$ and $\vec{n}_2$ and rearranging terms, we get:

$$\begin{aligned} \{\vec{r} - \vec{r}_1\} \bullet (\vec{n}_1 \times \vec{n}_2) = 0 \\ \{\vec{r} - \vec{r}_2\} \bullet (\vec{n}_1 \times \vec{n}_2) = 0 \end{aligned} \qquad \textbf{Eq. 9}$$

Thus, $\vec{r}$ lies in the plane containing both P$_1$ and P$_2$ and normal to the vector $\vec{n}_1 \times \vec{n}_2$. Assuming that $\vec{n}_1 \times \vec{n}_2$, we can then parameterize $\vec{r}$ as (see Figure 1):

$$\vec{r} = \vec{o} + a\frac{\vec{n}}{\|\vec{n}\|} \text{ and } \vec{n} = b\vec{n}_1 + (1-b)\vec{n}_2 \qquad \textbf{Eq. 10}$$
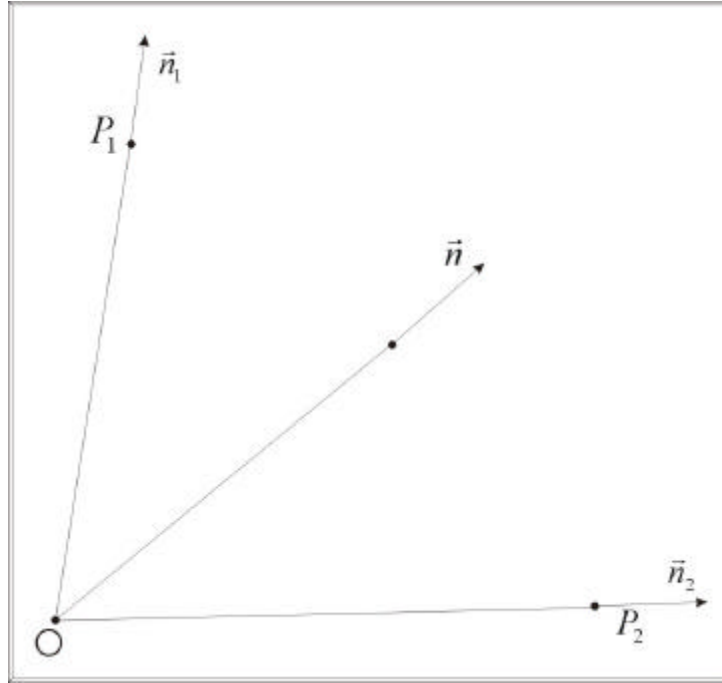

Figure 1. Plane containing $\vec{r}$ also contains the normal vectors to $\vec{r}$.

Where $\vec{o}$ is an arbitrary origin. Using Equation 10 to substitute for $\vec{r}$, $\vec{r}_1$ and $\vec{r}_2$ in Equation 8 and expanding, we get:

$$a\frac{\vec{n}_1 \times \vec{n}_2}{\|n\|}[1-b] = \vec{0} \qquad \textbf{Eq. 11}$$

From Equation 11, it follows that since $\vec{n}_1 \times \vec{n}_2$ cannot be zero (as per our assumption), λ is equal to β. In all of the subsequent analysis, we will use β, with the understanding that the parameterizations by λ and β are equivalent.

We now impose the following constraint on C:

$$\frac{d\vec{r}}{ds} \bullet \vec{n}(s) = 0 \qquad \textbf{Eq. 12}$$

where s is the arc length, and $\vec{n}(s)$ is the normal vector field. This condition requires the tangent vector to the curve at every point on the curve to be perpendicular to the normal to the surface,

effectively forcing the curve to lie on the specified surface S. It also requires the normal vector to the curve to be parallel to the normal vector to the surface. Substituting from Equation 10:

$$\frac{d\vec{r}}{ds} = \frac{d\boldsymbol{a}}{ds}\left(\frac{\vec{n}}{\|\vec{n}\|}\right) + \boldsymbol{a}\left(\frac{\|\vec{n}\|\dfrac{d\vec{n}}{d\boldsymbol{b}} - \vec{n}\dfrac{1}{\|\vec{n}\|}\left(\vec{n}\bullet\dfrac{d\vec{n}}{d\boldsymbol{b}}\right)}{\|\vec{n}\|^2}\right)\frac{d\boldsymbol{b}}{ds} \qquad \text{Eq. 13}$$

or, upon simplification and substitution of Equation 11 in Equation 12:

$$\frac{d\vec{r}}{ds}\bullet\vec{n}(s) = \frac{d\boldsymbol{a}}{ds}\|\vec{n}\| = 0 \qquad \text{Eq. 14}$$

which implies that:

$$\boldsymbol{a}(s) = \text{constant} \qquad \text{Eq. 15}$$

With Equation 15, Equation 10 reduces to that of a circular arc in the plane. The unit normal vector to any point on this arc is equal to the unit normal vector to the surface S at that point

To prove the converse, i.e. given a circular arc lying on the surface S with the unit normal vector field to the arc equal to the unit normal vector field of the surface, we write Equations 2 and 3 for a circular arc. Thus Equation 2 becomes:

$$(\vec{v} + \vec{w}\times(\vec{o} + R\vec{n}_1))\bullet\vec{n}_1 \geq 0 \qquad \text{Eq. 16}$$

and Equation 3 becomes:

$$(\vec{v} + \vec{w}\times(\vec{o} + R\vec{n}_2))\bullet\vec{n}_2 \geq 0 \qquad \text{Eq. 17}$$

where R is the radius of the circular arc. Expanding Equations 17 and 18 and forming a linear combination:

$$(\vec{v} + \vec{w}\times\vec{o})\bullet(\boldsymbol{l}\vec{n}_1 + (1-\boldsymbol{l})\vec{n}_2) \geq 0 \qquad \text{Eq. 18}$$

which is of the same form as Equation 1. Therefore, non-penetration at the points P_1 and P_2 implies non-penetration all along C. Note that this is similar to the proof of non-penetration along a circular

arc on a right circular cylinder, presented in Sinha et al. [1998]. This completes the proof of *Proposition 1*.

Lemma 1.1. *The curve C exists on surface S when:*

1. $(\vec{r}_1 - \vec{r}_2) \bullet (\vec{n}_1 \times \vec{n}_2) = 0$
2. *The intersection of S with the above plane is a circular arc.*

Proof. The above two conditions follow from *Proposition 1*. Curve C lies in a plane containing the points P_1 and P_2, as defined in Equation 9. C is also a circular arc on S, as shown in Equation 15.

Corollary 1.1. *The straight line* $l(\boldsymbol{l}) : \boldsymbol{l} \in [0,1] \rightarrow \boldsymbol{l}\vec{r}_1 + (1-\boldsymbol{l})\vec{r}_2$ *on a plane surface S satisfies Proposition 1.*

Corollary 1.2. *The great arc on a spherical surface S subtending an angle less than $\boldsymbol{p}$ satisfies Proposition 1.*

Corollary 1.3. *The straight vertical line parallel to the axis and the circular arc subtending an angle less than $\boldsymbol{p}$ on a right circular cylindrical surface S satisfies Proposition 1.*

Proof. Corollaries 1.1 through 1.3 are discussed and proved individually in Sinha et al. [1998]. Here, we show that they emerge as special cases of *Proposition 1*. As per the Proposition, the only possible segment on a planar surface (Corollary 1.1), with $\vec{n}_1$ equal to $\vec{n}_2$, is the circular arc with infinite radius, i.e. the straight line joining the points P_1 and P_2. The great arc on a spherical surface also satisfies *Proposition 1*. The possible segments on a right circular cylindrical surface are the vertical straight line and the circular arc. The subtended angle is required to be less than π to prevent $\vec{n}_1$ being parallel to $\vec{n}_2$.

Corollary 1.4. *The straight vertical line starting at the apex of a right conical surface S satisfies Proposition 1.*

Proof. Upon examination of a right circular conical surface, we see the straight line (or meridian lines) of the cone has its unit normal vector equal to the unit normal vector of the surface of the cone. Therefore, *Proposition 1* is satisfied. Note that circular arcs can be present on the cone, but do not satisfy *Proposition 1* because the unit normals along the arc are not equal to the unit normal vectors of the cone.

4. ARTICULATION IN ASSEMBLIES

The existence of an unconstrained degree of freedom will cause the generalized velocity space representation generated from the non-penetration conditions to be non-empty. Therefore, the velocity space can be analyzed to detect and classify unconstrained degrees of freedom.

4.1. Solving the Set of Non-Penetration Conditions for Instantaneous Articulation

Each primitive patch induces a non-penetration condition at each of its (finite) vertices. Since non-penetration must not occur at any point at any time, the inequalities for all the non-penetration conditions for the all the patches of a pair of bodies considered simultaneously form the linear program:

$$\vec{n}_i \bullet (\vec{v} + \vec{w} \times \vec{r}_i) \geq 0 \qquad i = 1 \ldots \text{Total numer of vertices in all patches} \qquad \textbf{Eq. 19}$$

where $\vec{v}$ and $\vec{w}$ are the relative velocities between the two parts, $\vec{n}_i$ is the normal at each vertex and $\vec{r}_i$ is the position of each vertex.

Since at any time, *all* the non-penetration conditions for all the parts must be satisfied, it is possible to solve all the inequalities for all the vertices of all the parts in the same linear program. This will result in a solution which is globally valid. Using a single linear program, it is possible to obtain all the instantaneous degrees of freedom for the assembly. If we write the non-penetration condition for a patch concisely as:

$$\mathbf{J}_{patch} \begin{bmatrix} \vec{v}_A - \vec{v}_B \\ \vec{w}_A - \vec{w}_B \end{bmatrix} \geq \vec{0} \qquad \textbf{Eq. 20}$$

where $\mathbf{J}_{patch}$ is the Jacobian for that particular patch. Then the non-penetration conditions for all the patches in a body-body contact pair formed between bodies A and B can be written as:

$$\mathbf{J}_{AB} \begin{bmatrix} \vec{v}_A - \vec{v}_B \\ \vec{w}_A - \vec{w}_B \end{bmatrix} = \begin{bmatrix} \mathbf{J}_{patch_1} \\ \mathbf{J}_{patch_2} \\ \vdots \\ \mathbf{J}_{patch_p} \end{bmatrix} \begin{bmatrix} \vec{v}_A - \vec{v}_B \\ \vec{w}_A - \vec{w}_B \end{bmatrix} \geq \vec{0} \qquad \textbf{Eq. 21}$$

where p is the number of patches in which this body-body pair participates. Using expressions such as Equation 22 written for all the body-body contact pairs in an assembly, we get:

$$\mathbf{J}_{assembly} \begin{bmatrix} \vec{v}_A \\ \vec{w}_A \\ \vec{v}_B \\ \vec{w}_B \\ \vdots \end{bmatrix} = \begin{bmatrix} \mathbf{J}_{AB} & -\mathbf{J}_{AB} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{J}_{BC} & -\mathbf{J}_{BC} & \cdots & \mathbf{0} \\ \vdots & \vdots & \vdots & \ddots & \vdots \end{bmatrix} \begin{bmatrix} \vec{v}_A \\ \vec{w}_A \\ \vec{v}_B \\ \vec{w}_B \\ \vdots \end{bmatrix} \geq \vec{0} \qquad \textbf{Eq. 22}$$

$\mathbf{J}_{assembly}$ is a complete representation of the assembly with instantaneous articulation. Solving this global simplex provides all the translational and angular velocities for all the body-body pairs simultaneously.

The simplex is 6N-6-dimensional (3 variables for translational velocity and 3 variables for angular velocity for each of N-1 ungrounded bodies in the assembly). Since the origin is a vertex of this high-dimensional space, this structure is also called a Polyhedral Convex Cone. Such structures have been studied extensively by Goldman and Tucker [1956] and others. Hirai and Asada [1993] used cones to describe the possible contact-preserving and contact-breaking motions between two polyhedral bodies.

4.2. A CAD Implementation for Instantaneous Articulation

Having established a theoretical framework for treating curved surface contacts in the previous sections, we now describe the system which extracts non-penetration conditions from the CAD models of parts in an assembly. A *contact graph* structure G can be used to represent the assembly. In the contact graph, parts are represented as nodes, and contacts between parts are represented as edges between the corresponding nodes. Edges between nodes are automatically derived by performing intersections between the nodes. Each part is scaled by a measure proportional to its bounding box dimensions, so that the result of the intersection is a regular solid (or a set of regular solids).

$$G = \{(A, B, I): A, B \in \text{Set of parts}; A \neq B; I = \text{Result of intersection between A and B}\} \qquad \textbf{Eq. 23}$$

The intersection information in each edge is examined for features that could indicate the presence of surface mating constraints. Each element of I can be thought of as a *constraint patch* on the mating surface between A and B. For a particular element, all boundary segments that do not satisfy *Proposition 1* are discretized into *primitive segments* (straight lines on planes, circular arcs and straight lines on a cylinder for constant z and constant θ respectively, and great arcs on a sphere), each of which satisfy *Proposition 1*. The set I can be partitioned as:

$$I = \underbrace{S_1 \cup S_2 \cup S_3 \cup \ldots \cup S_n}_{\text{finite } n} \qquad \textbf{Eq. 24}$$

where S_1 through S_n are a finite number of *primitive surfaces* (or portions thereof). On each S, there exists a finite set of boundary segments Ω:

$$\Omega = \{ \boldsymbol{s} : \boldsymbol{s} \in \text{Set of boundary segments of S} \}$$
$$\text{s.t. } \boldsymbol{s} \approx \boldsymbol{\hat{s}} = \underbrace{\boldsymbol{a}_1 \cup \boldsymbol{a}_2 \cup \ldots \cup \boldsymbol{a}_m}_{\text{finite } m} \qquad \textbf{Eq. 25}$$

where σ is either a primitive segment, or can be approximated as $\boldsymbol{\hat{s}}$ which is a union of a finite number of primitive segments α. Thus, the intersection set I now is composed of a finite number of primitive boundary segments.

The non-penetration condition for each end-point or vertex of each primitive boundary segment α is written as a constraint in the linear program. The linear space can now be described (or enumerated) using standard boundary enumeration techniques. Note that this technique will work only for those surfaces which satisfy *Proposition 1*. On other surfaces such as splines, non-penetration conditions would have to be written for every point on the surface.

Finding one solution to the simplex is easy; finding *all* solutions is a more difficult proposition. However, useful information can still be obtained by projecting the simplex on either the $\vec{v}$ or the $\vec{w}$ space. Such linear programming methods have previously been used by Mattikalli et al. [1994] to obtain solutions to the stability problem for assemblies.

Solutions are returned in the form of allowable instantaneous translational and angular velocities. Translational velocities of zero indicate that translation is constrained for that body-body pair. Angular velocities of zero indicate that rotation is constrained for that body-body pair.

4.3. Giving Feedback to the Designer

In order to completely describe all the possible relative motions of the assembly, it is necessary to completely describe the boundary of the polyhedral convex cone in 6N-6-dimensional space.

Useful feedback can be provided to the designer in the form of questions such as: "What degrees of freedom exist when the rotations of a particular part are constrained?" This question can be answered by adding $\vec{w} = \vec{0}$ for the body in question, to the set of constraints and evaluating the linear program. Other possible "what-if" analyses include grounding a part (i.e. setting $\vec{v} = \vec{0}$ and $\vec{w} = \vec{0}$ for that part) and obtaining the instantaneous degrees of freedom for all the other parts.

The space of allowed motions can be represented by the set-theoretic sum of the space of motions which preserve the contact ($J_{assembly}\vec{V} = \vec{0}$), and the space of motions which break the contact ($J_{assembly}\vec{V} > \vec{0}$), where $\vec{V}$ is the vector of generalized velocities:

$$S_{allowed} = S_{contact-preserving} + S_{contact-breaking} \qquad \textbf{Eq. 26}$$

S$_{contact-preserving}$ is the space of possible generalized velocity vectors which cause all contacts to be maintained, or:

$$S_{contact-preserving} = Nullspace(J_{assembly})$$

<div align="right">**Eq. 27**</div>

The basis vectors of the nullspace completely describe the possible contact-preserving motions. A singular value decomposition of **J**$_{assembly}$ is used to compute the nullspace.

Computing the boundary of the space of contact-breaking motions is a much harder problem. Avis [1994], Avis and Fukuda [1991, 1996], Motzkin et al. [1953], Bremner et al. [1996], Fukuda and Prodon [1996] have all proposed methods to enumerate the boundary of a polyhedral convex cone. However, time requirements for these methods quickly explode when confronted with cones of increasing dimensionality. Nemhauser and Wolsey [1988] show that the extreme ray membership problem for a cone is in NP. Similar conclusions are drawn by Avis [1998]. For a discussion of the complexity class of enumeration problems see Fukuda [1998].

Given a cone, it is possible to verify the feasibility of a given solution in polynomial time [Nemhauser and Wolsey, 1988]. Therefore, we propose the following heuristic to construct a finite set of feasible solutions from the geometry of the assembly:

1. For every primitive contact patch, pick the axis and a position on the axis - for planes, the normal to the plane; for cylinders and cones, the principal axis; for spheres, any axis; in addition, pick a second axis tangent to the direction of curve parameterization – for planes, an axis lying in the plane, for cylinders and cones, the axis of the base; for spheres, any axis perpendicular to the first axis.
2. See if a translational velocity along and/or a rotational velocity about these axes are feasible solutions to the set of non-penetration conditions.
3. Since there are a finite number of patches, therefore a finite number of axes, the number of possible feasible solutions will be finite. Note that this heuristic will fail to detect degrees of freedom that are not along axes of symmetry.

Introduction of domain-specific knowledge (geometric information, in this case), enables us to draw useful inferences about the space of allowed motions. The feasible solutions that emerge from this test, along with the nullspace of the cone, form a representation of the cone. Feasible solutions for a particular pair of parts can be combined to form joints between parts.

5. ILLUSTRATIVE EXAMPLES

5.1. Example 1 (Arm): Demonstration of the Framework

In order to illustrate the framework defined in the previous sections, we present an example assembly. The assembly in Figure 2 is a 4-part assembly, with three functional degrees of freedom. We choose to render the base immobile (i.e. we *ground* it), by constraining all the six degrees of freedom.
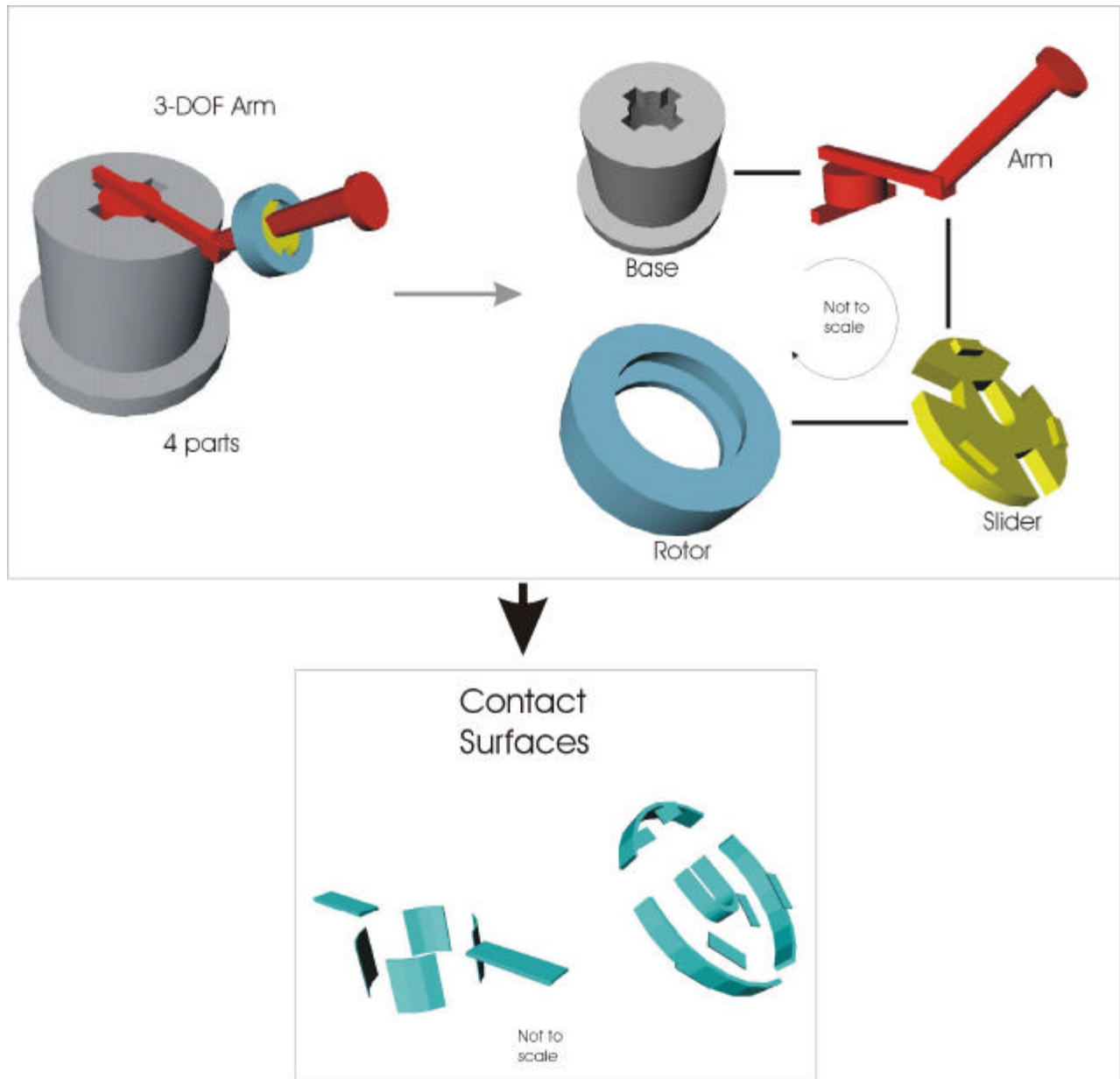

Figure 2. 4-part assembly with 3 degrees of freedom.

The system is implemented in C++, using ACIS as a solid modeller, and with Open Inventor for 3-D visualization. Contact analysis indicates that this is an open-chain assembly, with 9 linear-boundary planar contacts (base-to-arm, arm-to-slider and slider-to-rotor), 1 curved-boundary planar contact (base-to-arm), 8 partial cylindrical contacts (base-to-arm, arm-to-slider and slider-to-rotor). This generates a total of 81 inequalities, linear in 18 variables forming the relative translational and rotational velocities (assuming the base is grounded).
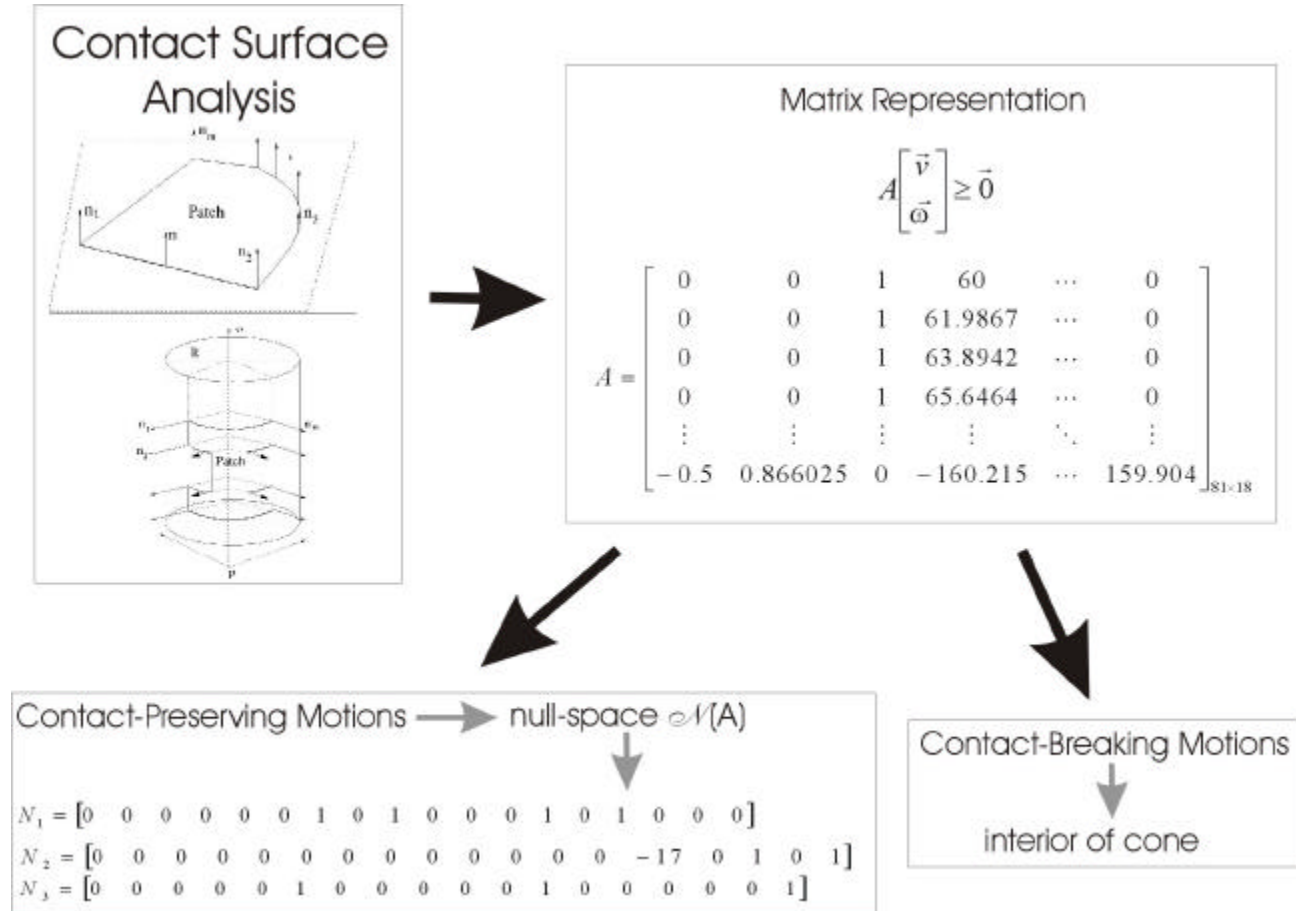

Figure 3. Formation of the Matrix Representation.

The 81 constraints define a polyhedral convex cone in 18-dimensional space (see Figure 3). The matrix representation can be partitioned into two: the nullspace and the interior boundary enumeration representation. Singular value decomposition returns 3 singular values, indicating that there are three completely unconstrained degree of freedom (and as a result, the nullspace is 3-dimensional). The nullspace basis $[N_1, N_2, N_3]$ is shown in Equation 29.

14

$$N_1 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 & 1 & 0 & 1 & 0 & 0 & 0 \end{bmatrix}$$
$$N_2 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -17 & 0 & 1 & 0 & 1 \end{bmatrix}$$
$$N_3 = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \qquad \textbf{Eq. 28}$$

where

$$N_i = \begin{bmatrix} \underbrace{v_x & v_y & v_z & W_x & W_y & W_z}_{\text{Arm}} & \underbrace{v_x & v_y & v_z & W_x & W_y & W_z}_{\text{Slider}} & \underbrace{v_x & v_y & v_z & W_x & W_y & W_z}_{\text{Rotor}} \end{bmatrix}$$

N_3

Nullspace basis vector N_1 indicates the presence of a translational degree of freedom for the slider and rotor with respect to the base. This freedom is at a 45° angle to the horizontal plane, indicated by the equal infinitesimal relative translational velocities in the x and z directions. Basis vector N_2 indicates that there is a relative instantaneous rotation of the rotor with respect to the base at a 45° angle to the horizontal plane. The [0 −17 0] translational component stems from the fact that the instantaneous rotation is not about the origin, but instead about the axis [1 0 1] through the point [18 0 1].

For the instantaneous velocity of the body to be zero at the point [18 0 1],

$$\vec{v}_{total} = \vec{v}' + \vec{w} \times \vec{p} = \vec{0} \qquad \textbf{Eq. 29}$$

or:

$$v' = \vec{p} \times \vec{w} = \begin{bmatrix} 0 & -17 & 0 \end{bmatrix} \qquad \textbf{Eq. 30}$$

where v′ is the translational component, namely [0 −17 0]; $\vec{w}$ is the instantaneous angular velocity component, namely [1 0 1]; and $\vec{p}$ is the locus of positions which satisfies Equation 30. Thus, we get an axis [1 0 1] passing through [18 0 1]. Thus, nullspace analysis describes the boundary of the cone.

Feasible solutions are constructed from the contact patches. The 18 primitive patches result in 216 candidate feasible solutions. Of these, four solutions are found to be valid. Of the 4 valid solutions, 3 are identical to the nullspace basis vectors. The fourth is [0 0 1 0 0 0 0 0 1 0 0 0 0 0 1 0 0 0], corresponding to the contact-breaking motion (vertical translation along the z-axis) between the base and the other 3 parts.
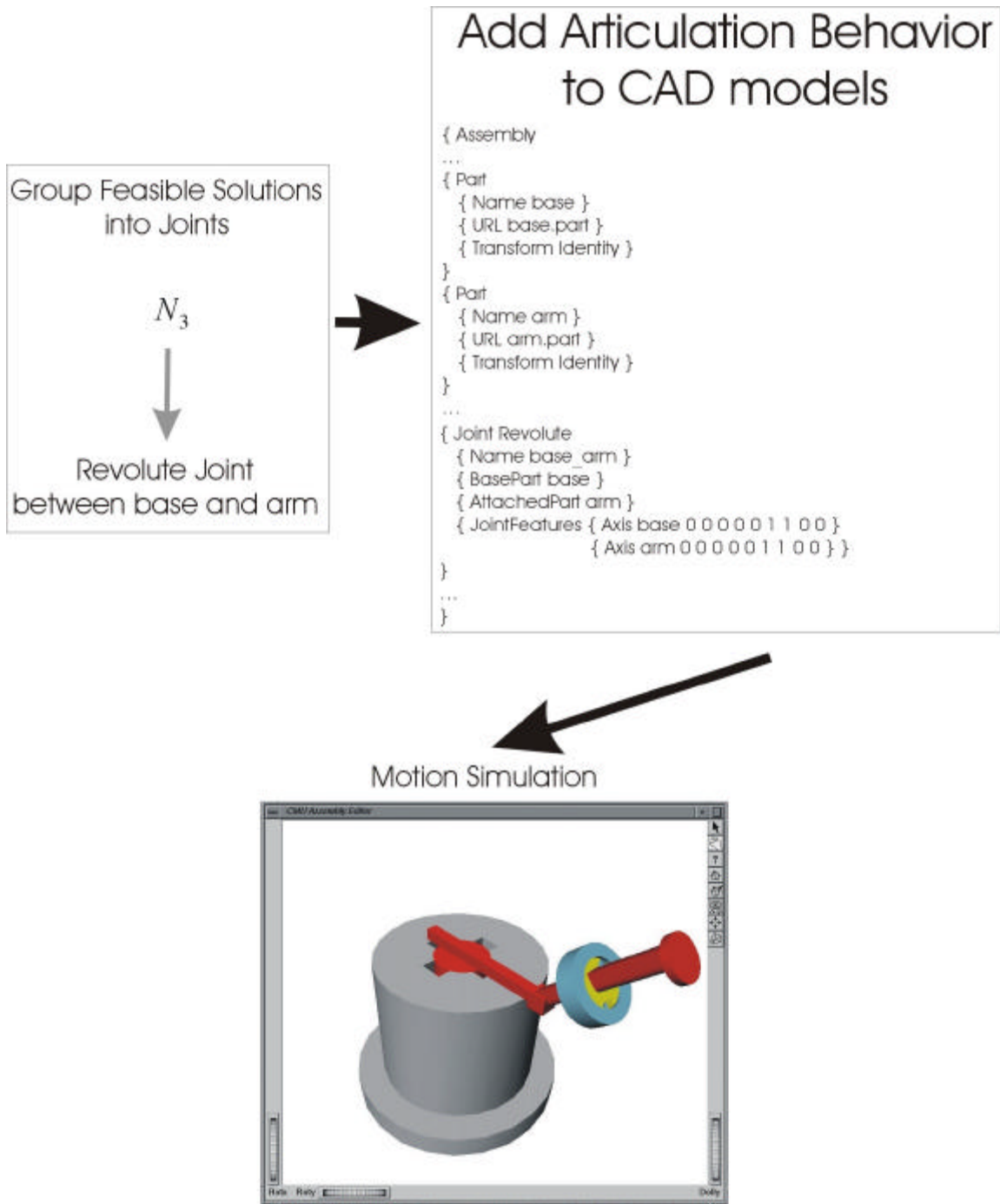
Figure 4. Mapping feasible solutions to assembly joints.

Once valid feasible solutions are available, it is possible to group them together to define joints between parts in the assembly. The two feasible solutions in this example result in a revolute joint being formed between the two subassemblies (see Figure 4). Joint information is added to the representation of the assembly. Such automatic detection of joints is useful when performing motion simulation.

16

5.2. Example 2 (1-DOF 4-Bar Linkage): Global Constraint Resolution

To verify the correctness of our implementation, we compared the results of our system against the analytical formulation for a 4-bar linkage. The example in this subsection is a 4-bar assembly with 1 degree of freedom (see Figure 5) with each bar having a joint axis-to-joint axis distance of 39 units. Contact analysis indicates that there are 8 curved-boundary planar contacts (2 between each pair of bars) and 4 complete cylindrical contact (1 between each pair of bars).
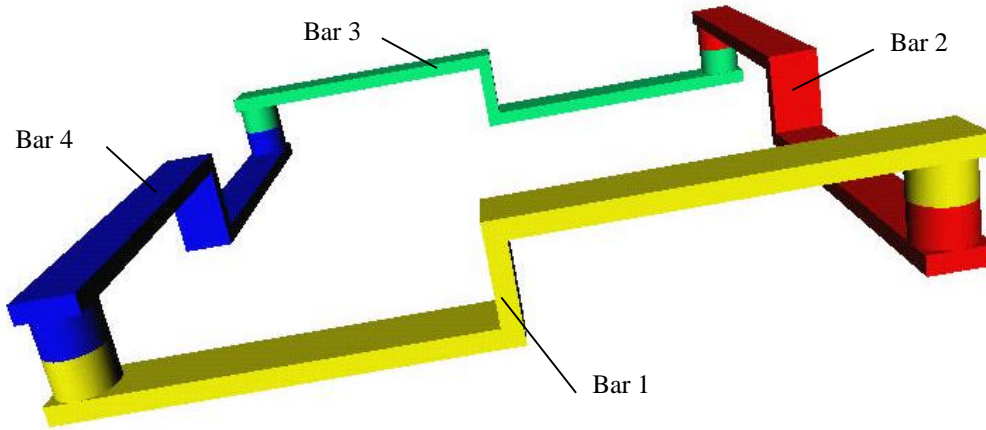

Figure 5. 4-Bar Linkage

This generates 460 inequalities in 18 relative translational and rotational velocity variables (assuming that Bar 1 is grounded). The inequality coefficient matrix A (Figure 3) has a rank of 17. Therefore, the nullspace N is 1-dimensional, indicating the presence of 1 completely unconstrained degree of freedom. A basis vector for the nullspace of A is:

$$N_1 = \begin{bmatrix} 0 & -39 & 0 & 0 & 0 & 1 & -39 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

where   **Eq. 31**

$$N_1 = \begin{bmatrix} \underbrace{v_x \quad v_y \quad v_z \quad W_x \quad W_y \quad W_z}_{Bar\,2} & \underbrace{v_x \quad v_y \quad v_z \quad W_x \quad W_y \quad W_z}_{Bar\,3} & \underbrace{v_x \quad v_y \quad v_z \quad W_x \quad W_y \quad W_z}_{Bar\,4} \end{bmatrix}$$

Removing one of the bars would have resulted in an open-chain assembly, with 3 more degrees of freedom. The closed-loop eliminates these extra degrees of freedom, and this is reflected in the 1-dimensional nullspace. Examining the nullspace, we see that for an unit instantaneous rotation of Bar 4 with respect to Bar 1 about the z-axis, Bar 2 undergoes a unit instantaneous rotation about the z-axis, but at a position of [39 0 0] (using Equation 29). Bar 3 undergoes an instantaneous pure translation along the x-axis of a magnitude 39 times that of the rotation.

The 4-bar linkage has been extensively studied by mechanism theorists. Its behavior is well known. From [Paul, 1979] (see pp. 233-234), we see that the "velocity-loop equation" written for our 4-bar linkage is:

$$L\vec{w}_4 - L\vec{w}_2 = \vec{0}$$
$$L\vec{w}_3 = \vec{0}$$

where L is the length of the bar, $\vec{w}_2$ is the angular velocity of Bar 2 with respect to Bar 1, $\vec{w}_3$ is the angular velocity of Bar 3 with respect to Bar 1, and $\vec{w}_4$ is the angular velocity of Bar 4 with respect to Bar 1. Thus, $\vec{w}_2$ is equal to $\vec{w}_4$ and $\vec{w}_3$ is zero. Since the links are rigid, for every $d\vec{q}$ rotation of Bar 2 and Bar 4, there must be a $Ld\vec{q}$ x-axis displacement of Bar 3. This is identical to the previous result. This validates our implementation of degree of freedom extraction.

6. DISCUSSION

In current design practice, when a device which contained parts with curved surfaces is considered, reasoning about its behavior is usually done by analytical methods that are difficult to implement, or by approximating all the curves as piecewise planar surfaces [Mattikalli et al., 1994]. The approximation is made on the basis of past experience (i.e. selecting the number of planes to represent a curved surface is a heuristic). Such an approximation propagates throughout the design process - part and assembly designers must ensure that the approximation does not constrain the degrees of freedom in the device; analysts must ensure that assemblies which are stable are not modeled as unstable; tolerancing specialists must account for the fact that the approximated surfaces may have modeled tolerances that are different from the desired specifications; process engineers may discover problems of fit and/or interferences in the model when actually no problem occurs (and vice versa). Assembly process engineers, tool engineers and operators may notice that some assembly tasks are impossible to perform due to this approximation.

Even with an acceptable approximation, it is difficult to propagate the physical behavior through the different aspects of the design process. Even in a collaborative setting, when a part or assembly used in a previous design project was reused in a new design project, all that was known *a priori* was the geometry of the part or assembly. Only the CAD model of the part or assembly persisted across designs. No physical model-related information about the part or assembly was reused. In addition, feature-based design of complex devices requires verification of the fact that actual kinematic behavior matches the required behavior.

The above discussion indicates a need for three capabilities that will support all aspects of the design process - creation of the ability to exactly reason with curved surfaces; the ability to obtain kinematics from assembly geometry; and the ability to package kinematic behavior into the representation of a part or an assembly.

A change in the geometry of a part or a change in the relative positions of parts necessitates a complete regeneration of the models of the mechanical behavior of the assembly. If an acceptable model that mimics the mechanics behavior of an artifact can be extracted from the geometry of the artifact, and if mechanics models are merged when the corresponding geometry is merged, then the designer will be able to: create new designs out of existing artifacts or parts and reuse portions or all of older designs in new design projects. In addition, the mechanical models will ensure physically correct behavior for the integrated artifact; this implies that any automatically generated simulation will be faithful to the models. Therefore, the designer will not have to recreate a simulation each time a new product is designed from previously created artifacts or parts.

At the conceptual level, the designer knows the type and behavior of the joints in the concept design. However, the geometry still needs to be defined. At the preliminary design stage, the geometric information is defined, and the relative positions and orientations of components in the assembly are specified. Following this stage, component and joint representations are enriched to achieve a final refinement of the geometry.

In order to generate assembly or disassembly plans for such assemblies, the designer needs to take articulation information into consideration. However, current methods of representing articulation are restricted to systems that require complete specification by the user [ADAMS, 1998] or are feature recognition based [Rajan et al., 1997]. The former are open to incorrect input by the user resulting in illegal articulation behavior. The latter do not account for incomplete geometry and incidental contacts.

Our method is able to handle incomplete curved geometry, while at the same time resolving global (i.e. multi-part) constraint interactions. Linear algebra-based constraint models are derived directly from CAD models, and then converted into articulation representations suitable for assembly planning and motion simulation.

7. CONCLUSIONS AND FUTURE WORK

This research forwards the state-of-the-art in the following ways:

Exact Treatment of Curved Geometry. Previous work on the application of linear programming techniques to the contact mechanics between two surfaces was restricted to handling planar surfaces only (see [Mattikalli et al, 1994] and [Baraff and Mattikalli, 1993]). Curved surfaces were handled by approximating them as planar facets, or by using graphical methods. This result was extended to curved surfaces whose boundaries possess certain properties. The vast majority of manufactured parts have contacts that possess such boundaries. Therefore, the work presented here allows a designer to analyze and design devices that contain parts (or components) that in turn contain a broad class of curved surfaces.

Modeling Support For Kinematics Behavior. Solving the contact conditions at the boundary will enable the designer to obtain the instantaneous degrees of freedom of the device. Using generate-and-test methods and reasoning about the geometry of the artifact will reveal the motion limits to the degrees of freedom. Such information, when combined with information about the mass distribution and with friction models, will allow the designer to determine the stability of the device, handle articulation during assembly planning, during synthesis of part geometry, as well as during the operation of the device.

Accurate and Automatic Simulation Synthesis. Simulation of the operation of the electromechanical device requires kinematic and dynamic information. Since each component incorporates such information, high-fidelity simulation can be created and executed with minimal user interaction.

Interesting research issues remain with regard to the handling of model uncertainty and in the quantification of the effect of model approximations, such as the discretization of curved planar contact boundaries, on the final result of the articulation analysis.

ACKNOWLEDGEMENTS

REFERENCES

1. ADAMS Virtual Prototyping Software, Mechanical Dynamics, http://www.adams.com, 1998.
2. Avis, D., "Computational Experience with the Reverse Search Vertex Enumeration Algorithm", *Optimization Methods and Software*, 1998, to appear.
3. Avis, D., "A C Implementation of the Reverse Search Vertex Enumeration Algorithm," in *RIMS Kokyuroku 872*, H. Imai, Ed., Kyoto University, May 1994.
4. Avis, D.; and Fukuda, K., "Reverse Search for Enumeration," *Discrete Applied Math*, **6:**21-46, 1996.
5. Avis, D.; and Fukuda, K., "A Pivoting Algorithm for Convex Hulls and Vertex Enumeration of Arrangements and Polyhedra", *Proceedings of the ACM 7th Symposium on Computational Geometry*, 1991, pp. 98-104.
6. Baraff, David; and Mattikalli, Raju, "Impending Motion Direction of Contacting Rigid Bodies", *Technical Report CMU-RI-TR-93-15*, The Robotics Institute, Carnegie Mellon University, Pittsburgh, PA 15213.
7. Bremner, D.; Fukuda, K.; and Marzetta, A., "Primal-Dual Methods for Vertex and Facet Enumeration", *Proceedings of the ACM 13th Symposium on Computational Geometry*, IFOR, ETH Zurich, December 1996.
8. Fukuda, K., "Class ENP: An Extension of Class NP $\cap$ co-NP", http://www.ifor.math.ethz.ch/ifor/staff/fukuda/ENP_home/ENP_note.html, a short note in HTML, 1998.
9. Fukuda, K.; and Prodon, A., "Double Description Method Revisited", in Combinatorics and Computer Science, M. Deza, R. Euler and I. Manoussakis, (Eds.), *Lecture Notes in Computer Science*, Springer-Verlag, 1996, Vol. 1120, pp. 91-111.
10. Goldman, A.J.; and Tucker, A.W., "Polyhedral Convex Cones", in Linear Inequalities and Related Systems, H.W. Kuhn and A.W.Tucker, (Eds.), *Annals of Math Studies*, Princeton University Press, Princeton, NJ, 1956, Vol. 39, pp. 19-39.
11. Gray, Alfred, Modern Differential Geometry of Curves and Surfaces with Mathematica, CRC Press, New York, USA, 1998.
12. Hirai, Shinichi; and Asada, Haruhiko, "Kinematics and Statics of Manipulation Using the Theory of Polyhedral Convex Cones", *International Journal of Robotics Research*, 1993, **12**(5):434-447.
13. Mattikalli, Raju; Baraff, David; and Khosla, Pradeep, "Finding All Gravitationally Stable Orientations of Assemblies", *Proceedings of the IEEE International Conference of Robotics and Automation*, San Diego, May 1994, pp. 251-257.
14. Mattikalli, R.; and Khosla, P.K., "Analysis of Restraints to Translational and Rotational Motion from the Geometry of Contact", DE-Vol. 39, Issues in Design Manufacture/Integration, *Proceedings of the ASME Winter Annual Meeting*, Atlanta, GA, December 1-6, 1991, pp. 65-71.
15. McCleary, John, Geometry from a Differentiable Viewpoint, Cambridge University Press, New York, USA, 1997.
16. Motzkin, T.S.; Raiffa, H.; Thompson, G.L.; and Thrall, R.M., "The Double Description Method", in Contributions to the Theory of Games Vol. 2, H.W. Kuhn and A.W.Tucker, (Eds.), Princeton University Press, Princeton, NJ, 1953.

17. Nemhauser, G.L.; and Wolsey, L.A., <u>Integer and Combinatorial Optimization</u>, Wiley Interscience, New York, USA, 1988.

18. Ohwovoriole, M.S.; and Roth, B., "An Extension of Screw Theory", *ASME Journal of Mechanical Design*, **103**:725-735, 1981.

19. Paul, B., <u>Kinematics and Dynamics of Planar Machinery</u>, Prentice-Hall Inc., Englewood Cliffs, NJ, USA, 1979.

20. Rajan, V.N.; and Nof, S.Y., "Minimal Precedence Constraints for Integrated Assembly and Execution Planning", *IEEE Transactions on Robotics and Automation*, Vol. 12, No. 2, April 1996, pp. 175-186.

21. Rajan, Venkat N.; Lyons, Kevin W.; and Sreerangam, Raj, "Generation of Component Degrees of Freedom from Assembly Surface Mating Constraints", *Proceedings of the 1997 ASME Design Engineering Technical Conference*, September 14-17, 1997, Sacramento, California.

22. Sinha, R.; Paredis, C.J.J.; Gupta, S.K.; and Khosla, P.K., "Capturing Articulation in Assemblies from Component Geometry", *Proceedings of the 1998 ASME Design Engineering Technical Conference*, September 13-17, 1998, Atlanta, Georgia.