Augmenting a Nominal Assembly Motion Plan with a Compliant Behavior [1]

Gordon Dakin Robin Popplestone

COINS Technical Report 91-06

May 21, 1991

Laboratory for Perceptual Robotics
Department of Computer and Information Science
A305, Graduate Research Center
University of Massachusetts
Amherst, MA 01003

Abstract

A methodology is presented whereby a nominal assembly trajectory for an assembly operation, computed from kinematic constraints alone, is augmented with a fine-motion strategy synthesized through uncertainty and force analyses. Insertion clearances and size tolerances are introduced into the assembly part models in parallel with the manual synthesis of a perturbed nominal trajectory *in contact space. The selection of small clearances, and in turn, small insertion angles allows us to linearize contact space about discrete points in the nominal trajectory. Contact states are represented as affine spaces in a generalized C-space of model error annd pose variables. The feasibility of proposed command velocities to be executed in the presence of position, control, and model error is determined through an uncertainty analysis technique based upon the forward-projection of convex polytopes in contact space. Our approach further automates the so-called "manual" methods of motion planning with uncertainty.*

[1]Preparation of this paper was supported by grant number N00014-84-K-0564, from the Office of Naval Research.

1 Introduction

In the *2-phase approach* to assembly motion planning, a nominal plan is first selected or derived with limited regard to the effects of sensor and control error, and then provisions are made to account for uncertainty. These provisions might entail refining the nominal plan [20][25] or augmenting the plan with constraints upon the initial conditions [2] or applied forces [26]. The 2-phase approach differs from the *LMT* methods [4][11][12][21], in which the command motions themselves are derived to accommodate uncertainty. A drawback of the LMT approach is its time complexity of $O(2^{2^n})$ in the number of plan steps [7]. An advantage of the 2-phase approach is that goal configurations and nominal trajectories may be derived in phase 1 from geometric constraints alone [1][17][18][19][23].

The approaches to motion planning with uncertainty that have been referred to by Latombe [15] as the "manual methods" were first developed for deriving applied force constraints to prevent jamming and wedging in the context of the 2-dimensional *peg-in-hole problem* [24][26]. Ohwovoriole [22] extended peg-in-hole force analysis to multiple insertion tasks in 3 dimensions. As demonstrated in [6], it may be necessary to restrict the path of the assembly to a single, designated path in contact space, if jamming is to be prevented using a single applied wrench that satisfies the jamming-avoidance constraints in all anticipated contact states. Caine [5] developed designer tools for manually selecting a trajectory of assembly configurations, and for deriving applied wrench constraints to prevent jamming and the breaking of contact while traversing the specified contact states.

The methodology presented in this paper combines the 2-phase and LMT approaches to assembly motion planning, while adopting and partially automating the "designer" techniques of the manual methods. We are chiefly concerned with the second phase of the 2-phase approach. Input consists of a nominal mating trajectory derived by the high-level assembly motion planner KA3 [18] from the feature symmetries of geometric models that permit zero clearance at the insertion sites. Small clearances and tolerances are added to these models, giving rise to local C-spaces of perturbations from discrete points in the nominal trajectory. The small insertion angles permitted by the narrow clearances warrant the linearization of these C-spaces, whose boundaries are characterized as sets of *linearized C-surfaces* in a generalized C-space of model error and pose variables. Linear programming-based tools are em-

ployed to confirm the existence and adjacency of polytopic contact states in the contact state lattice. Command trajectories for traversing each polytopic C-space are then synthesized, while accommodating the uncertainty in the initial pose, control velocity, and geometric models of the assembly parts. Finally, applied force constraints are generated for maintaining jam–free contact in the specified states. The resulting fine motion strategy consists of an initial target pose, a sequence of command velocities, and the applied wrench constraints required for a jam-free traversal of a specified sequence of contact states.

Like the "designer" approach of Caine [5], our methodology does not exclude human participation in the development of a fine-motion strategy. Tools for generating and verifying a fine-motion plan permit one to juggle design variables which include trajectory perturbations, clearances, and tolerances. The policy of forming fine-motion plans in parallel with selection of clearances and tolerances is consistent with the view that a product should be designed with the feasibility of its assembly in mind [27]. As in Donald's approach to fine-motion synthesis in the presence of model error, our method relies on forward-projecting uncertainty regions within a generalized C-space augmented with model error dimensions. Whereas Donald's *error detection and recovery strategies (EDRs)* are developed by augmenting existing fine motion plans with contingency strategies that accommodate erroneous transitions due to model error, the method described in this paper permits the synthesis of fine-motion strategies "from scratch" in generalized C-space. Once the possibility of an undesired transition is detected through forward-projection, the designer can modify the command velocities, clearances, or tolerances, until the assembly trajectory is restricted to a single path of contact states.

2 Enumeration of Primitive Contacts

We assume that a high-level assembly planner ([18]) has supplied a nominal trajectory, a sequence of connected line segments in $\Re^6$ describing the motion of a *moving part* relative to a *stationary part.* We are also provided with geometric models of the parts, along with a set of *critical points*, i.e., poses at which the set of contacting surface features change (see figure 1a). The contacts that can arise in the presence of insertion clearances are de-

termined by examining the contacts occurring at the critical points prior to introducing clearances to the models (see figure 1b). Any contact between the surface features of two polyhedral objects can be represented as a combination of *primitive contacts*: vertex-face, face-vertex, or edge-edge pairs (where each pair denotes moving and stationary part features, resp.) involving convex vertices and edges. The n primitive contacts $P_1, \ldots, P_n$ associated with a critical point are enumerated by detecting coincident features in the clearance-free models. The vertex-face contacts consist of all pairs (V, F) where convex vertex V lies within the polygon of face F. Face-vertex contacts are similar. Edge-edge contacts involve pairs (E_1, E_2) of convex edges whose line segments intersect.

Small clearances are now added to the models at the insertion sites by "shrinking" various dimensions of the moving part, stationary part, or both. In figure 2a, for example, clearances are introduced by receding a hole wall along its negated normal by a distance dc. Model dimension errors are represented by the displacment $\mathbf{dr} = [dr_x dr_y dr_z]^T$ of V from its model to actual position, and scalar dr_w, the displacement of F along its normal. These model error variables are subject to tolerances $|dr_i| \leq \epsilon_i,\ i \in w, x, y, z$.

3 Linearized C-Surfaces

Each primitive contact P_i that can occur around a critical point in the trajectory is characterized by a *linearized C-surface* in $\Re^{t+6}$, consisting of the tangent hyperplane to the actual C-surface of moving part poses associated with P_i. Following Donald [10], we represent model error variables as additional *dofs* in a generalized C-space. Linearized C-surfaces for vertex-plane and edge-edge contacts subject to model error are derived as follows.

Figure 2a shows a vertex V of the moving part and a face F of the stationary part, separated by a clearance dc. Model error variables $\mathbf{dr}$ and dr_w displace V and F to their real positions. A perturbation of the moving part from the critical point is denoted by a twist $\mathbf{dX} = [\mathbf{dx}^T\ \ \delta\mathbf{x}^T]^T$ with differential translation and rotation vectors $\mathbf{dx}$, $\delta\mathbf{x}$ (resp.). The placement of V against F is expressed by stating that V and F have the same ordinate along F's normal $\mathbf{n}$:

$$(\mathbf{p} + \mathbf{r} + \mathbf{dr} + \mathbf{dx} + \delta\mathbf{x} \times (\mathbf{r} + \mathbf{dr})) \cdot \mathbf{n} = (\mathbf{p} + \mathbf{r} - (dc + dr_w)\mathbf{n}) \cdot \mathbf{n} \qquad (1)$$

where $\mathbf{r}$ is the vector of displacement from the moving part's origin $\mathbf{p}$ to V. After eliminating the nonlinear term $(\delta\mathbf{x} \times \mathbf{dr}) \cdot \mathbf{n}$, equation (1) may be expressed:

$$\begin{bmatrix} \mathbf{n} \\ \mathbf{r} \times \mathbf{n} \\ 1 \\ \mathbf{n} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{dx} \\ \delta\mathbf{x} \\ dr_w \\ \mathbf{dr} \end{bmatrix} = -dc \tag{2}$$

which describes a hyperplane in $\Re^{10}$, whose normal is the left hand vector.

In figure 2b, edges E_1 and E_2 are subject to model error displacements $\mathbf{dr}_1$ and $\mathbf{dr}_2$ (resp.), and the edges are parallel to $\mathbf{v}_1$ and $\mathbf{v}_2$. Equating the ordinates of E_1 and E_2 along their mutual perpendicular $\mathbf{v}_1 \times \mathbf{v}_2$ and dropping nonlinear terms yields:

$$\begin{bmatrix} \mathbf{v}_1 \times \mathbf{v}_2 \\ \mathbf{r}_1 \times (\mathbf{v}_1 \times \mathbf{v}_2) \\ \mathbf{v}_1 \times \mathbf{v}_2 \\ -\mathbf{v}_1 \times \mathbf{v}_2 \end{bmatrix} \cdot \begin{bmatrix} \mathbf{dx} \\ \delta\mathbf{x} \\ \mathbf{dr}_1 \\ \mathbf{dr}_2 \end{bmatrix} = \mathbf{v}_1 \times \mathbf{v}_2 \cdot \mathbf{dc} \tag{3}$$

where $\mathbf{r}_1$ is the displacement from the moving part's origin $\mathbf{p}$ to point $\mathbf{q}_1$ on E_1, and $\mathbf{q}_1$ coincides with point $\mathbf{q}_2$ on E_2 prior to introducing clearance vector $\mathbf{dc}$. For parallel edge-edge contacts, a distinct $\mathbf{r}$ does not exist and $\mathbf{v}_1 \times \mathbf{v}_2 = \mathbf{0}$, so we employ values of $\mathbf{r}$ and $\mathbf{v}_1 \times \mathbf{v}_2$ associated with target assembly poses (see section 5). Equation (3) describes a hyperplane in $\Re^{12}$, whose normal consists of the lefthand vector. In general, linearized C-surfaces are constructed in a $(t+6)$-dimensional C-space, where t is the number of model dimensions subject to size tolerance.

4 Convex Regions in C-space

In addition to a linearized C-surface equation, our representation of a primitive contact's linearized C-surface must also include an inequality for each C-surface boundary arising from a real surface boundary. In figure 3a, the moving part is situated at the critical point at which its base is aligned with the outside rim face F of a hole in the stationary part. An edge E_1 of the moving part's base may contact one of the rim edges E_2, so we characterize

this contact by a linearized C-surface, represented by the hyperplanar equality formulated in equation (3). For E_1 to contact E_2, however, E_1's vertex V must lie on the left side of hole wall G_1. This constraint may be represented by an inequality, bounded by equation (2). We thus establish a vertex-face C-surface as a barrier beyond which the edge-edge C-surface does not extend. Erecting a similar inequality constraint to represent the finite extent of E_2, the primitive contact between E_1 and E_2 is represented by a constraint set containing the original equality, plus two inequalities, in $\Re^{t+6}$.

Each primitive contact P_i is thus characterized as a convex, hyperplanar polytope H_i in $\Re^{t+6}$. Each polytope H_i is represented by a *constraint set* S_i consisting of a single equation and a number of inequalities in $\Re^{t+6}$, as one might represent a polygon in $\Re^2$. Primitive contact constraint sets may also include artificial constraints involving imaginary faces, such as G_2 in figure 3a, which splits the concave primitive contact between V and F into two convex primitive contacts.

The set of nonoverlapping configurations surrounding a critical point, denoted by set "$LEGAL$", is also concave but can be constructed as a union of convex volumes $\mathcal{L}_1, \ldots \mathcal{L}_m$ in $\Re^{t+6}$. The planar moving part in figure 3b, for example, moves in a 3−dimensional C-space whose subset $LEGAL$ is concave, owing to the convex hole rims. $LEGAL$ may be partitioned into four convex volumes in $\Re^3$, corresponding to the four possible assignments of the moving part vertices V_1 and V_2 to the convex regions above or below their adjacent hole rims. Each volume $\mathcal{L}_i$ may be represented by artificial constraints confining each of V_1 and V_2 to its assigned convex region, plus constraints prohibiting overlap between a vertex and a hole wall. Assuming all concavities in $LEGAL$ arise from convex hole rims of the stationary part, we can generally characterize $LEGAL$ as a union of convex volumes $\mathcal{L}_i$, where each $\mathcal{L}_i$ corresponds to an assignment of the moving part's "rim vertices" to halfspaces on one side or the other of their adjacent stationary part hole rims.

The construction of $LEGAL$ as a union of convex volumes is performed for 3−dimensional assemblies as follows. The part variations permitted by the tolerances and the rigidity of the parts permit some n partitionings of rim vertices to the halfspaces above or below their adjacent rim planes. Given n above/below vertex assignments, we construct $LEGAL$ as a union of n convex C-space volumes $\mathcal{L}_1, \ldots, \mathcal{L}_n$. Each volume $\mathcal{L}_i$ is represented by a constraint set $\mathcal{S}_i$ of inequalities in $\Re^6$. For each above/below vertex assignment, we construct volume $\mathcal{L}_i$ by restricting the moving part vertices associated

with the two sets of projections to the halfspaces above or below their rim planes, as the assignment dictates. The inequalities that make up a convex volume's constraint set S_i include inequalities that restrict the vertices to their assigned halfspaces, as well as inequalities to prevent overlap between the surfaces of the parts.

Refer again to figure 3a, which shows a vertex V of the moving part adjacent to a hole with rim edges E_2 and E_3, and rim face F. The moving part lies at a critical point placing V in the plane of face F. To construct an inequality constraint set S_1 for a $LEGAL$ subvolume $\mathcal{L}_1$ that places V above the plane of F, we first calculate the linearized C-surface equation formulated in equation (2) for the V-against-F contact, and express it as an inequality (negating it if necessary) that restricts V to the halfspace above F. Constraints are also added for other rim vertices' above/below placements. Other inequality constraints in S_1 include (among others) inequalities formulated by equation (3) for the E_1-against-E_2 contact and the E_1-against-E_3 contact.

To construct a constraint set S_2 for a C-space volume $\mathcal{L}_2$ that places V below the plane of F, we express the V-against-F equation (2) as an inequality (negating it if necessary) restricting V to the halfspace below F. This constraint is added to similar constraints for other rim vertices' above/below placements. The constraint which keeps V, submerged in the hole, from overlapping with the interior hole walls G_1 is formulated by equation (2). A constraint preventing V's overlap with its other adjacent wall, plus edge-against-edge constraints similar to those described above, are also included in S_2.

5 Generating Paths in Contact Space

Once the primitive contacts $P_1, \ldots, P_n$ that may occur around a critical point are characterized by their hyperplanar polytopes $H_1, \ldots, H_n$ and corresponding constraints sets $S_1, \ldots, S_n$, we can determine through linear programming which combinations of primitive contacts might occur simultaneously. A *contact state* $C = \{P'_1, \ldots, P'_k\}$ is a k-element subset of $\{P_1, \ldots, P_n\}$, such that $P'_1, \ldots, P'_k$ can coexist in C-space without overlap, and without necessarily the presence of an additional P'_l. The contact states $C_1, \ldots, C_m$, plus the empty set $\emptyset$ and $\{P_1, \ldots, P_n\}$, form a lattice under the relation of set inclu-

sion [14]. The region in C-space represented by a contact $C = \{P'_1, \ldots, P'_k\}$ is the intersection $(\bigcap_{i=1}^{k} H'_i) \cap LEGAL$ of the primitive contacts' polytopes with $LEGAL$. We shall refer to a contact state C as an *n-dimensional state* if its tangent space in $\Re^{t+6}$ has rank n, or equivalently, if the vector space spanned by the H'_is' hyperplanar normals has rank $t + 6 - n$.

To generate tentative paths of traversal in the contact space around a critical point, we require computational tools to (1) decide if a set of primitive contacts forms a contact state, and (2) identify its adjacent states. As for (1), k primitive contacts $P'_1, \ldots, P'_k$ can coexist without overlap iff $(\bigcap_{i=1}^{k} H'_i) \cap \mathcal{L}_j \neq \emptyset$ for some convex component $\mathcal{L}_j$ of $LEGAL$. This decision is performed as a linear programming feasibility test involving constraint sets $S'_1, \ldots S'_k$ of the k primitive contacts and the constraint set S_j of $\mathcal{L}_j$. We moreover determine whether $\{P'_1, \ldots, P'_k\}$ can coexist without any additional contact P'_l by determining if $\{P'_1, \ldots, P'_k, \bar{P}'_l\}$ can coexist without overlap, where imaginary contact $\bar{P}'_l$ represents a slightly separated P'_l contact. Regarding (2), we enumerate the contact states C' adjacent to a given state $C = \{P'_1, \ldots, P'_k\}$ by adding or subtracting a P'_i and checking that the new combination is a valid contact state.

The decision procedures described above may be utilized to generate candidate sequences of traversable contact states around each successive critical point. As seen in the contact state graph in figure 4a, for example, the assembly motion begins in free space $C_0 = \emptyset$. The 1-point contact states surrounding the first critical point include an edge-edge contact state C_1 and a vertex-plane contact state C'_1. After verifying the legality of these states, we choose (say) C_1 and enumerate its adjacent $2-$point contact states, which include C_2 and C'_2, and so on. The final contact state in the selected sequence must be shared by the contact state lattice associated with the next critical point, where it will serve as the initial contact state in the next sequence of contact states.

6 Command Velocity Synthesis

Once a path of contact states $C_0, \ldots, C_m$ is selected for traversing the contact space around a critical point in the nominal trajectory, we specify *target poses* in some or all of the contact states. Each target pose $\mathbf{dX}_i \in \Re^6$ is a perturbation of the assembly's pose away from the critical point. In the case

of the first critical point, the initial target pose $\mathbf{dX_0}$ serves as the approach position in free space to which the moving part is (say) visually servoed. Every $\mathbf{dX_i}$ thereafter is the goal point associated with a contact state transition involving the establishment of an additional primitive contact, i.e., a transition to a lower-dimensional contact state (see figure 5a). The target poses $\mathbf{dX_0}, \ldots, \mathbf{dX_l}$ associated with a contact state sequence $C_0, \ldots, C_m$ (with $l \leq m$) comprise a *perturbed nominal trajectory* (PNT) in the contact state lattice surrounding the critical point.

A human designer may specify the target pose in a contact state by constraining any *dofs* in the assembly configuration left unconstrained by the primitive contacts. As shown in n figure 4b, the target pose in a contact state involving a single edge-edge contact is specified by supplying five "virtual" vertex-face contacts involving imaginary faces formed by perturbing existing faces by various distances d_i. Constraining the pose of an assembly to a single point $\mathbf{dX_i}$ in $\Re^6$ generally requires $6-k$ such artificial constraints, where k is the rank of the space spanned by the primitive contacts' normals. As shown in figure 5a, each adjacent pair of target poses $\mathbf{dX_i}, \mathbf{dX_{i+1}}$ gives rise to a unit command velocity $\mathbf{v_i}$ parallel to $\mathbf{dX_{i+1}} - \mathbf{dX_i}$. Instead of specifying the PNT in terms a sequence of target poses, the designer may alternatively supply the sequence of command velocities directly, together with an initial target pose $\mathbf{dX_0}$. Beginning with $i = 0$, each sequential pair $\mathbf{dX_i}, \mathbf{v_i}$ determines the next target pose $\mathbf{dX_{i+1}}$. When the fine-motion plan executes, recognition of each new contact via force sensing triggers the command velocity associated with the new contact state.

After selecting a PNT within the chosen sequence of contact states, we must ensure that trajectory deviations arising from position and control error will not give rise to undesired contact state transitions. At the start of the assembly operation, the pose of the moving part is confined to an uncertainty region R_0 in $(t+6)$-dimensional free space. As shown in figure 5a, we "verify" a candidate PNT by recursively forward-projecting the current uncertainty region R_i within contact state S_i, to obtain the uncertainty region R_{i+1} in the next contact state S_{i+1} that contains a target pose. Each R_i sweeps out a forward-projection volume F_i centered around command velocity $\mathbf{v_i}$ and (due to velocity uncertainty) expanding laterally to $\mathbf{v_i}$.

We employ a linear-programming feasibility test to determine if a forward-projection F_i intersects any undesired primitive contact's polytope H_j. A positive test result shows that for *some* universe allowed by the part dimen-

sion tolerances, an undesired contact state transition *might* occur, due to position and control error. Accordingly, the PNT, the clearances, or the tolerances must be modified. Tightening the tolerances permits the forward-projections to bypass the unwanted contacts more easily. But since tighter tolerances increase the cost of manufacturing the parts [13], it is desirable to obtain a fine motion strategy with as wide tolerance ranges as possible. If F_i is shown not to intersect any undesired primitive contacts' polytopes in $\Re^{t+6}$, then the command velocity $\mathbf{v}_i$ defined by the sequential target poses $\mathbf{dX}_i$ and $\mathbf{dX}_{i+1}$ is valid. To verify subsequent command velocities, the process repeats by forward-projecting the uncertainty region R_{i+1} from S_{i+1} to S_{i+2}, and so on.

As seen in figure 5a, the forward-projection F_i of a 3−dimensional region R_i is bounded by planes, such as Γ_g, formed by translating a 1−dimensional facet g along some deviating velocity $\mathbf{w}_g$. A 1−dimensional facet g gives rise to such a bounding plane of F_i iff g does not pierce the interior of R_i when displaced along $\pm\mathbf{w}_g$. Only 6 of the 12 1−dimensional facets so qualify. We assume that the maximum angular deviation of a control velocity $\mathbf{w}$ from the command velocity $\mathbf{v}$ is available as a function $\epsilon_v(\mathbf{u})$ of the lateral deviation direction $\mathbf{u} \perp \mathbf{v}$ (see figure 5b). When calculating g's deviating velocity $\mathbf{w}_g$, we choose $\mathbf{u} \perp \mathbf{v}, g$, as shown in figure 5c. The resulting constraint plane Γ_g defines a halfspace which contains all possible trajectories from g. Γ_g and five similarly-derived constraint planes define the boundary of F_i, together with the constraint plane of the new primitive contact in state C_{i+1}, as well as the 3 "upper" 2−dimensional facets of R_i (see figure 5a).

We now address the process of forward-projecting pose uncertainty regions in a generalized, $(t+6)$−dimensional space. In figure 6a, a polytopic uncertainty region R_0 projects to a lower-dimensional C-surface that varies in position orthogonally to its normal, with tolerance limits $\pm dr$ imposed upon its deviation from the nominal position. As the C-surface varies, the polytope's projected image varies, generating polytopes that are similar, with parallel corresponding facets. This observed similarity and parallellism is preserved when the projected polytopes themselves are in turn projected onto subsequent variable C-surfaces. In general, any two projected images of the original polytope onto successive, variable C-surfaces must always have the same number of facets. Most importantly, there exists a 1−to−1 correspondence between the two images' facets, and *this is true for all in-tolerance positions of the C-surface.* The 1−to−1 correspondence stems from the fact

that the two sets of facets are multiply-projected images of the same subset of facets of the original polytope.

By performing forward-projections in generalized C-space, all the possible variations in the projected images arising from model error variations may be computed "in parallel", using a single, higher-dimensional facet to encode the possible variations in a facet of the projected uncertainty region. The preceding discussion indicates that such a generalized facet takes on all values of the tolerance variables, i.e., the facet will not vanish for some model error parameterization.

Figure 6b shows a 3-dimensional, generalized C-space with 2 spatial dimensions (x and y) and a single tolerance dimension (T). The uncertainty region R_0 is a rectangular block, bounded above and below by planes Γ^+ and Γ^-, which represent tolerance limits. R_0 is projected along command velocity $\mathbf{v}_0$ onto the planar surface S of a primitive contact. Two of R_0's 1-dimensional facets (g_1 and g_2) project to boundary facets (g_1' and g_2') of the projected image R_1. As discussed above, g_1' and g_2' take on all values of T. Since R_1 is entirely bounded by g_1', g_2', Γ^+, and Γ^-, and because Γ^+ and Γ^- are invariant for all R_is, the boundary of R_1 (and subsequent projected regions R_i) is defined by multiply-projected images of a subset of R_0's pose uncertainty-related (vertical) facets, and by Γ^+ and Γ^-.

The preceding observations extend to $\Re^{t+6}$. We assume that the initial uncertainty region in free space is a hyperrectangle R_0 in $\Re^{t+6}$. R_0 can be characterized by the $2t + 12$ $(t + 5)-$dimensional facets of its border, or by the $2t + 12$ $\Re^{t+6}$-halfspaces associated with those facets. R_0 has $2t$ facets of dimension $t + 5$ associated with plus-and-minus tolerance bounds, and 12 facets of dimension $t + 5$, as well as 60 facets of dimension $t + 4$, associated with initial pose error bounds. When R_0 is projected onto the $(t + 5)-$dimensional affine space associated with the first primitive contact, a subset of the $(t + 4)-$dimensional facets project to border facets of R_0's projected image R_1. If R_1 itself is projected onto the affine space of a second primitive contact, some of its $(t + 3)-$dimensional facets project to border facets of the projected image R_2, and so on.

As demonstrated previously, the boundary facets of an uncertainty region R_i always consist of multiply-projected images of a subset of the original R_0's pose uncertainty-related facets, plus R_0's $2t$ hyperplanes associated with tolerance limits. Since R_0's pose uncertainty-related facets are just the facets

of a 6-dimensional pose uncertainty hyperrectangle (lifted to $\Re^{t+6}$), it is convenient to construct a boundary model for a hyperrectangle in $\Re^6$, to serve as a data structure to hold affine space descriptions of the pose uncertainty-related facets of R_0. As a decreasing subset of the hyperrectangle's facets are successively projected onto the hyperplanes of new C-surfaces, the facets' affine spaces are updated to describe their projected images. The boundary model consists of an adjacency graph of 0− to 5−dimensional facets, with associated affine space descriptors. The affine space of a facet f is represented by a vector subspace basis in $\Re^{t+6}$, denoted by "*basis*(f)", and a representative point in $\Re^{t+6}$, denoted by "*point*(f)".

The calculation of R_0's projected image R_1 in the first contact state entails (1) determining which $(t+4)$−dimensional, pose uncertainty-related facets g of R_0 project to an outer boundary facet g' of R_1, and then (2) updating the affine space representation of g to describe its projected image g'. More generally, we can project an uncertainty region R_i from a $(t+6-k)$−dimensional contact state into its image R_{i+1} in a $(t+5-k)$-dimensional contact state by (1) determining *which* $(t+4-k)$−dimensional facets g of R_i project to an outer boundary facet g' of R_{i+1}, and then (2) updating the affine space representation of g to describe its projected image g'.

We present an algorithm at the end of this section which forward-projects a polytopic uncertainty region R_i from a $(t+6-k)$−dimensional contact state C_i onto its image R_{i+1} in a $(t+5-k)$-dimensional contact state C_{i+1}. In addition to calculating R_{i+1}, the algorithm constructs the forward-projection F_i of R_i, i.e., the volume in $\Re^{t+6}$ swept out by R_i in the presence of velocity uncertainty. The projection F_i is centered around command velocity $\mathbf{v}_i$, which points from the target pose $\mathbf{dX}_i$ in C_i to the target pose $\mathbf{dX}_{i+1}$ in C_{i+1}.

The variables and data structures employed are: k linearized C-surface descriptors $(\mathbf{N}_1, D_1), \ldots (\mathbf{N}_k, D_k)$ associated with the $(t+6-k)$-dimensional contact state's primitive contacts, where $(\mathbf{N}_i \in \Re^{t+6}, D_i \in \Re)$ denotes the equation $\mathbf{N}_i \cdot \mathbf{dX} = D_i$ in generalized C-space; the descriptor $(\mathbf{N}_{k+1}, D_{k+1})$ associated with the new primitive contact in the $(t+5-k)$-dimensional state; $REGION_i$, a list of $\Re^{t+6}$−halfspaces of the form $\mathbf{N} \cdot \mathbf{dX} \leq D$ whose intersection with the C-surface hyperplanes of $(\mathbf{N}_1, D_1), \ldots, (\mathbf{N}_k, D_k)$ defines uncertainty region R_i; $FORWARD_i$, a list of $\Re^{t+6}$−halfspaces of the form $\mathbf{N} \cdot \mathbf{dX} \leq D$ whose intersection with the C-surface hyperplanes of

$(\mathbf{N}_1, D_1), \ldots (\mathbf{N}_k, D_k)$ defines forward-projection volume F_i; $FACES_i$, a list which contains the $(t+5-k)$−dimensional facets of R_i; and $EDGES_i$, a list which contains the $(t+4-k)$−dimensional facets of R_i. Prior to the first forward-projection, lists $FACES_0$ and $EDGES_0$ are intialized to contain all the pose-uncertainty-related, $(t+5)$− and $(t+4)$−dimensional facets (resp.) of R_0. $REGION_0$ is initialized to hold the $\Re^{t+6}$-halfspaces associated with each facet in $FACES_0$. The subroutines used in the algorithm are described as follows:

- *projdir*$(g, \mathbf{v}_i)$: computes the direction $\mathbf{w}_g$ toward which a facet g of R_i projects when R_i is forward-projected along command velocity $\mathbf{v}_i$. As illustrated in figures 5b and 5c, $\mathbf{w}_g$ is derived as the weighted vector sum $(sin\ \epsilon_v(\mathbf{u}))\mathbf{u}+(cos\ \epsilon_v(\mathbf{u}))\mathbf{v}$, where $\mathbf{u} \perp \mathbf{v}, basis(g), \mathbf{N}_1, \ldots, \mathbf{N}_k$.
- *exterior*$(g, \mathbf{w}_g)$: a predicate which returns true iff facet g of R_i, whose projection direction is $\mathbf{w}_g$, projects to an exterior boundary facet of R_{i+1}. g is a $(t+4-k)$−dimensional facet of R_i adjacent to two $(t+5-k)$−dimensional facets f_1, f_2 of R_i, and f_1 and f_2 confine R_i to two halfspaces $\mathbf{N}_1 \cdot \mathbf{dX} \leq D_1$ and $\mathbf{N}_2 \cdot \mathbf{dX} \leq D_2$ (resp.). This predicate returns true iff $sign(\mathbf{w}_g \cdot \mathbf{N}_1) \neq sign(\mathbf{w}_g \cdot \mathbf{N}_2)$.
- *perp*$(\{\mathbf{u}_1, \ldots, \mathbf{u}_n\})$: returns the $t+6-n$ vectors in $\Re^{t+6}$ which span the vector space orthogonal to $span\{\mathbf{u}_1, \ldots, \mathbf{u}_n\}$.
- *halfspace*$(\mathbf{N}, D, \mathbf{p})$: if $\mathbf{N} \cdot \mathbf{p} \leq D$ then the halfspace $\mathbf{N} \cdot \mathbf{dX} \leq D$ (i.e., its descriptors $\mathbf{N}, D$) is returned. Otherwise, halfspace $-\mathbf{N} \cdot \mathbf{dX} \leq -D$ is returned.
- *projpoint*$(\mathbf{p}, \mathbf{w}, \mathbf{N}, D)$: returns the point at which the line formed by point $\mathbf{p}$ and vector $\mathbf{w}$ intersects the hyperplane $\mathbf{N} \cdot \mathbf{dX} = D$.
- *solvesys*$(\mathbf{N}_1, D_1, \ldots, \mathbf{N}_n, D_n)$ returns a point in $\Re^{t+6}$ satisfying a possibly-underdetermined system $\mathbf{N}_i \cdot \mathbf{dX} = D_i, i = 1 \ldots n$.
- *top*$(H, \mathbf{v}_i)$: a predicate which returns true iff $(t+5-k)$−dim. facet f of R_i, associated with halfspace H of R_i, is one of the "top" facets of R_i with respect to a "downward" command velocity $\mathbf{v}_i$ (eg., facet f in figure 5a), signifying that the forward projection of f passes through the interior of R_i. This predicate returns true iff $\mathbf{v}$ points into halfspace H.

```
Procedure Forward-Proj  /*Forward-project pose uncertainty region R_i*/
  Initialize FORWARD_i, REGION_{i+1}, FACES_{i+1}, EDGES_{i+1} to empty
  For each facet g in EDGES_i do
    w_g ← projdir(g, v_i)
    If exterior(g, w_g)
      then
        /*Build (t + 5 − k)−dim affine space Γ_g swept out by g*/
        basis(Γ_g) ← basis(g) ∪ {w_g};  point(Γ_g) ← point(g)
        /*Derive halfspace descriptors N_g, D_g for Γ_g*/
        N_g ← perp(basis(Γ_g) ∪ {N_1, ..., N_k});  D_g ← N_g · point(g)
        Add halfspace(N_g, D_g, p_{i+1}) to FORWARD_i and REGION_{i+1}
        /*Update g's affine space to represent its projected image*/
        point(g) ← projpoint(point(g), w_g, N_{k+1}, D_{k+1})
        basis(g) ← perp({N_g, N_1, ..., N_{k+1}})
        Add g to FACES_{i+1}
        For each (t + 3 − k)−dim facet g' adjacent to g do
          Add g' to EDGES_{i+1} (if not already present)
        Endfor
    Endif
  Endfor
  For each facet g' in EDGES_{i+1} do
    /* Compute g''s affine space from its adjacent facets*/
    Find g''s 2 adjacent facets g_1, g_2 in FACES_{i+1}
    basis(g') ← perp({N_{g_1}, N_{g_2}, N_1, ..., N_{k+1}})
    point(g') ← solvesys((N_{g_1}, D_{g_1}), (N_{g_2}, D_{g_2}), (N_1, D_1), ..., (N_{k+1}, D_{k+1}))
  Endfor
  /*Seal off the "top" of the forward-projection volume*/
  For each halfspace (N, D) in REGION_i do
    If top(N, D, v_i)
      then
        Add (N, D) to FORWARD_i
    Endif
  Endfor
  /*Seal off the "bottom" of the forward-projection volume*/
  Add halfspace(N_{k+1}, D_{k+1}, p_i) to FORWARD_i
Endprocedure
```

The forward-projection volume computed above as list $FORWARD_{i+1}$ is submitted to a linear programming feasibility subroutine, together with any undesired primitive contact's hyperplanar polytope constraint set. A positive result indicates that the undesired contact can occur in the presence of uncertainty. In that event, the command velocities, clearances, and/or tolerances require modification.

7 Applied wrench constraints

After choosing a path of contact states to be traversed during an assembly operation, we must ensure that jamming (static friction) will not occur in any of the selected states, and that the contacts will not be broken once they are established. Figure 7a shows a point mass m sliding on a surface whose friction cone of reaction forces has half-angle $\tan^{-1}\mu$, where μ is the coefficient of friction. If the force $\mathbf{F}_{app}$ applied to m is directed toward the surface but does not point into the friction cone, the component of $\mathbf{F}_{app}$ unopposed by reaction force $\mathbf{F}_r$ generates an acceleration $\ddot{x}$ of m. If $\mathbf{F}_{app}$ points into the friction cone, the mass sticks on the surface.

We shall assume that the direction of the moving part's velocity in $\Re^6$ is sufficiently constrained so that the direction of the moving part's velocity at any contact point $\mathbf{p}_i$ will be very close to the unit velocity $\mathbf{v}_i$ associated with the moving part's perturbed nominal velocity. This assumption, together with our assumption of quasistatic conditions, enables us to express the reaction force $\mathbf{f}_i$ at each contact as a function of a single variable, the normal force f_i:

$$\mathbf{f}_i = f_i(\mathbf{n}_i - \mu\mathbf{v}_i) \tag{4}$$

where $\mathbf{n}_i$ is the contact normal (see figure 7b).

Following Caine [5], the applied wrench (force and torque) imparted to the moving part in a $k-$point contact state can be expressed

$$\begin{array}{rcl} \mathbf{F} & = & \sum_{i=1}^{k} f_i(-\mathbf{n} + \mu\mathbf{v}_i) + \delta\mathbf{v}_i \\ \mathbf{M} & = & \sum_{i=1}^{k} f_i\mathbf{r}_i \times (-\mathbf{n} + \mu\mathbf{v}_i) + \mathbf{r}_i \times \delta\mathbf{v}_i \end{array} \tag{5}$$

where $\mathbf{r}_i$ is the vector from the origin of the moving part to the contact at $\mathbf{p}_i$, and $0 < \delta \ll 1$ is associated with acceleratory sliding, while $-1 \gg \delta > 0$ represents the jamming condition.

7.1 Jamming avoidance in 1-point contact states

If $k = 1$, the applied wrench (5) can be written:

$$\begin{aligned} \mathbf{F} &= f_1\mathbf{A}_1 + \delta\mathbf{V} \\ \mathbf{M} &= f_1\mathbf{B}_1 + \delta\mathbf{W} \end{aligned} \tag{6}$$

where $\mathbf{A}_1 = -\mathbf{n}_1 + \mu\mathbf{v}_1$, $\mathbf{B}_1 = \mathbf{r}_1 \times (-\mathbf{n}_1 + \mu\mathbf{v}_1)$, $\mathbf{V} = \mathbf{v}_1$, $\mathbf{W} = \mathbf{r}_1 \times \mathbf{v}_1$. Figure 7a shows the 3-dimensional space of applied forces $\mathbf{F}$. Line $\ell_1 = \{\mathbf{F} \mid \mathbf{F} = f_1\mathbf{A}_1, f_1 \in \Re\}$ contains the applied forces associated with nonacceleratory sliding. The forces related to jamming lie in region $R = \{\mathbf{F} \,|\mathbf{F} = f_1\mathbf{A}_1 + \delta\mathbf{V}, f_1, \in \Re^+, \delta \in \Re^-\}$, which is bordered by ℓ_1. Any constraint plane Γ containing ℓ_1 but not R divides $\Re^3$ into two halfspaces, one of which contains R. Jamming can be prevented by confining $\mathbf{F}$ to the halfspace which contains $\mathbf{V}$. More generally, the subspace $\mathcal{S}$ of applied wrenches associated with nonacceleratory sliding, and the sticking region $\mathcal{R}$, may be expressed in $\Re^6$ as

$$\mathcal{S} = \left\{ \begin{pmatrix} \mathbf{F} \\ \mathbf{M} \end{pmatrix} \,\middle|\, \begin{pmatrix} \mathbf{F} \\ \mathbf{M} \end{pmatrix} = f_1 \begin{pmatrix} \mathbf{A}_1 \\ \mathbf{B}_1 \end{pmatrix}, f_1 \in \Re \right\} \tag{7}$$

$$\mathcal{R} = \left\{ \begin{pmatrix} \mathbf{F} \\ \mathbf{M} \end{pmatrix} \,\middle|\, \begin{pmatrix} \mathbf{F} \\ \mathbf{M} \end{pmatrix} = f_1 \begin{pmatrix} \mathbf{A}_1 \\ \mathbf{B}_1 \end{pmatrix} + \delta \begin{pmatrix} \mathbf{V} \\ \mathbf{W} \end{pmatrix}, f_1 \in \Re^+, \delta \in \Re^- \right\} \tag{8}$$

We can employ any 5-dimensional constraint hyperplane $\mathcal{H} \in \Re^6$ that contains $\mathcal{S}$, but not $\mathcal{R}$, to divide applied wrench space into two regions, one of which contains the sticking region $\mathcal{R}$ (see figure 8a). Such a hyperplane $\mathcal{H}$ may be specified by selecting five independent vectors in $\Re^6$ that include $[\mathbf{A}_1^T\mathbf{B}_1^T]^T$ but are independent of $[\mathbf{V}^T\mathbf{W}^T]^T$. The five vectors form a basis for the vector space comprising $\mathcal{H}$ and determine $\mathcal{H}$'s normal in $\Re^6$, which is computable via the Gram-Schmidt procedure.

7.2 Jamming avoidance in 2-point contact states

If $k = 2$, the applied wrench (5) can be written:

$$\begin{aligned} \mathbf{F} &= f_1\mathbf{A}_1 + f_2\mathbf{A}_2 + \delta\mathbf{V} \\ \mathbf{M} &= f_1\mathbf{B}_1 + f_2\mathbf{A}_2 + \delta\mathbf{W} \end{aligned} \tag{9}$$

where $\mathbf{A}_i = -\mathbf{n}_i + \mu\mathbf{v}_i$, $\mathbf{B}_i = \mathbf{r}_i \times (-\mathbf{n}_i + \mu\mathbf{v}_i)$ $(i = 1, 2)$, $\mathbf{V} = \mathbf{v}_1 + \mathbf{v}_2$, and $\mathbf{W} = \mathbf{r}_1 \times \mathbf{v}_1 + \mathbf{r}_2 \times \mathbf{v}_2$. As seen in figure 8b, vectors $\mathbf{A}_1$ and $\mathbf{A}_2$ span a plane $\Gamma \in \Re^3$, which contains the applied forces associated with nonacceleratory sliding. The forces related to jamming lie in a 3-D region $R = \{\mathbf{F} \mid \mathbf{F} = f_1\mathbf{A}_1 + f_2\mathbf{A}_2 + \delta\mathbf{V},\ f_1, f_2 \in \Re^+, \delta \in \Re^-\}$. As a constraint plane, Γ divides $\Re^3$ into two halfspaces, one of which contains R. Jamming can be prevented by confining $\mathbf{F}$ to the halfspace which contains $\mathbf{V}$. More generally, the subspace $\mathcal{S}$ of applied wrenches associated with nonacceleratory sliding, and the sticking region $\mathcal{R}$, may be expressed in $\Re^6$ as

$$\mathcal{S} = \left\{ \begin{pmatrix} \mathbf{F} \\ \mathbf{M} \end{pmatrix} \;\middle|\; \begin{pmatrix} \mathbf{F} \\ \mathbf{M} \end{pmatrix} = \left(\sum_{i=1}^{2} f_i \begin{pmatrix} \mathbf{A}_i \\ \mathbf{B}_i \end{pmatrix} \right), f_i \in \Re \right\} \tag{10}$$

$$\mathcal{R} = \left\{ \begin{pmatrix} \mathbf{F} \\ \mathbf{M} \end{pmatrix} \;\middle|\; \begin{pmatrix} \mathbf{F} \\ \mathbf{M} \end{pmatrix} = \left(\sum_{i=1}^{2} f_i \begin{pmatrix} \mathbf{A}_i \\ \mathbf{B}_i \end{pmatrix} \right) + \delta \begin{pmatrix} \mathbf{V} \\ \mathbf{W} \end{pmatrix} f_i \in \Re^+, \delta \in \Re^- \right\} \tag{11}$$

Any 5-dimensional constraint hyperplane $\mathcal{H} \in \Re^6$ that contains $\mathcal{S}$, but not $\mathcal{R}$, divides applied wrench space into two regions, one containing the sticking region $\mathcal{R}$. $\mathcal{H}$ is chosen by selecting five independent vectors in $\Re^6$ which include $[\mathbf{A}_1^T \mathbf{B}_1^T]^T$ and $[\mathbf{A}_2^T \mathbf{B}_2^T]^T$ but are independent of $[\mathbf{V}^T \mathbf{W}^T]^T$. The five vectors determine $\mathcal{H}$'s normal in $\Re^6$, which is computable via the Gram-Schmidt procedure.

7.3 Jamming avoidance in 3-point contact states

If $k = 3$, the applied wrench (5) can be written:

$$\begin{array}{rcl} \mathbf{F} & = & (\sum_{i=1}^{3} f_i\mathbf{A}_i) + \delta\mathbf{V} \\ \mathbf{M} & = & (\sum_{i=1}^{3} f_i\mathbf{B}_i) + \delta\mathbf{W} \end{array} \tag{12}$$

where $\mathbf{A}_i = -\mathbf{n}_i + \mu\mathbf{v}_i$, $\mathbf{B}_i = \mathbf{r}_i \times (-\mathbf{n}_i + \mu\mathbf{v}_i)$ $(i = 1, 3)$, $\mathbf{V} = \sum_{i=1}^{3} \mathbf{V}_i$, and $\mathbf{W} = \sum_{i=1}^{3} \mathbf{r}_i \times \mathbf{v}_i$. Generalizing the discussions of sections 7.1 and 7.2 to 3-point contacts, we can express the subspace $\mathcal{S}$ of applied wrenches associated with nonacceleratory sliding, and the sticking region $\mathcal{R}$ as sets in $\Re^6$:

$$\mathcal{S} = \left\{ \begin{pmatrix} \mathbf{F} \\ \mathbf{M} \end{pmatrix} \;\middle|\; \begin{pmatrix} \mathbf{F} \\ \mathbf{M} \end{pmatrix} = \sum_{1=1}^{3} f_i \begin{pmatrix} \mathbf{A}_i \\ \mathbf{B}_i \end{pmatrix}, f_i \in \Re \right\} \tag{13}$$

$$\mathcal{R} = \left\{ \begin{pmatrix} \mathbf{F} \\ \mathbf{M} \end{pmatrix} \middle| \begin{pmatrix} \mathbf{F} \\ \mathbf{M} \end{pmatrix} = \left(\sum_{1=1}^{3} f_i \begin{pmatrix} \mathbf{A}_i \\ \mathbf{B}_i \end{pmatrix} \right) + \delta \begin{pmatrix} \mathbf{V} \\ \mathbf{W} \end{pmatrix}, f_i \in \Re^+, \delta \in \Re^- \right\} \tag{14}$$

If a 5-dimensional hyperplane $\mathcal{H} \in \Re^6$ contains $\mathcal{S}$, but not $\mathcal{R}$, then $\mathcal{H}$ divides the applied wrench space into two regions, one of which contains the sticking region $\mathcal{R}$. $\mathcal{H}$ is determined by five independent vectors in $\Re^6$ that include $[\mathbf{A}_i^T \mathbf{B}_i^T]^T (i = 1 \ldots 3)$ but are independent of $[\mathbf{V}^T \mathbf{W}^T]^T$. $\mathcal{H}$'s normal in $\Re^6$ is then computed by the Gram-Schmidt procedure.

7.4 Maintaining a 1-point contact

According to equation (6), line $\ell_1 = \{\mathbf{F} \,|\mathbf{F} = f_1 \mathbf{A}_1, f_1 \in \Re\}$ contains the applied forces associated with nonacceleratory sliding ($\delta = 0$) in a 1-point contact state. The contact remains unbroken as long as the normal force f_1 is positive. As shown in figure 8c, any constraint plane Γ_1 that contains the origin but not ℓ_1 divides ℓ_1 into two rays: one is related to contact, and the other is related to separation. More generally, any hyperplane $\mathcal{H}_1 \in \Re^6$ that does not contain $[\mathbf{A}_1^T \mathbf{B}_1^T]^T$ divides the 1-dimensional subspace $\mathcal{S}$ of equation 7 into two regions, one of which contains all the applied wrenches associated with the breaking of contact. $\mathcal{H}_1$ may be specified by any 5 independent vectors in $\Re^6$ that do not contain $[\mathbf{A}_1^T \mathbf{B}_1^T]^T$ in their range space.

7.5 Maintaining a 2-point contact

From equation (9), the applied forces associated with nonacceleratory sliding ($\delta = 0$) and unbroken contact ($f_1, f_2 > 0$) in a 2-point contact state lie within a 2-D angular sector $s = \{\mathbf{F} \,|\mathbf{F} = f_1 \mathbf{A}_1 + f_2 \mathbf{A}_2, f_1, f_2 \in \Re^+\}$, bordered by lines $\ell_1 = \{\mathbf{F} \,|\mathbf{F} = f_1 \mathbf{A}_1, f_1 \in \Re\}$ and $\ell_2 = \{\mathbf{F} \,|\mathbf{F} = f_2 \mathbf{A}_2, f_2 \in \Re\}$. As shown in figure 8d, any pair of constraint planes Γ_1 and Γ_2 that contain ℓ_2 and ℓ_1 (resp.), but not ℓ_1 and ℓ_2 (resp.), may be employed to restrict $\mathbf{F}$ to s. More generally, the applied wrenches associated with 2-point contact maintenance are contained in a 2-dimensional angular sector in $\Re^6$:

$$\mathcal{S} = \left\{ \begin{pmatrix} \mathbf{F} \\ \mathbf{M} \end{pmatrix} \middle| \begin{pmatrix} \mathbf{F} \\ \mathbf{M} \end{pmatrix} = \sum_{i=1}^{2} f_i \begin{pmatrix} \mathbf{A}_i \\ \mathbf{B}_i \end{pmatrix} f_i \in \Re^+ \right\} \tag{15}$$

Any pair of constraint hyperplanes $\mathcal{H}_1, \mathcal{H}_2 \in \Re^6$ that contain $[\mathbf{A}_2^T \mathbf{B}_2^T]^T$ and

$[\mathbf{A}_1^T\mathbf{B}_1^T]^T$ (resp.) but not $[\mathbf{A}_1^T\mathbf{B}_1^T]^T$ and $[\mathbf{A}_2^T\mathbf{B}_2^T]^T$ (resp.) can be used to restrict the applied wrench to $\mathcal{S}$, with each $\mathcal{H}_i$ enforcing $f_i > 0$. $\mathcal{H}_1$ may be specified by any 5 independent vectors in $\Re^6$ that contain $[\mathbf{A}_2^T\mathbf{B}_2^T]^T$ but not $[\mathbf{A}_1^T\mathbf{B}_1^T]^T$ in their range space, and $\mathcal{H}_2$ may be specified similarly.

7.6 Maintaining a 3-point contact

The results of sections 7.4 and 7.5 generalize to 3-point contact states as follows. The applied wrenches associated with maintaining all contacts are contained in a 3-dimensional cone in $\Re^6$:

$$\mathcal{S} = \left\{ \begin{pmatrix} \mathbf{F} \\ \mathbf{M} \end{pmatrix} \middle| \begin{pmatrix} \mathbf{F} \\ \mathbf{M} \end{pmatrix} = \sum_{i=1}^{3} f_i \begin{pmatrix} \mathbf{A}_i \\ \mathbf{B}_i \end{pmatrix} f_i \in \Re^+ \right\} \tag{16}$$

Three constraint hyperplanes $\mathcal{H}_1, \mathcal{H}_2, \mathcal{H}_3 \in \Re^6$ are employed to restrict the applied wrench to $\mathcal{S}$. To enforce $f_1 > 0$, $\mathcal{H}_1$ must contain $[A_2^T B_2^T]^T$ and $[A_3^T B_3^T]^T$, but not $[A_1^T B_1^T]^T$. $\mathcal{H}_1$ may be specified by selecting any 5 independent vectors in $\Re^6$ that do not contain $[\mathbf{A}_2^T\mathbf{B}_2^T]^T$ or $[\mathbf{A}_3^T\mathbf{B}_3^T]^T$ in their range space. Similarly, $\mathcal{H}_2$ (for $f_2 > 0$) must contain $[A_1^T B_1^T]^T$ and $[A_3^T B_3^T]^T$, but not $[A_2^T B_2^T]^T$, while $\mathcal{H}_3$ (for $f_3 > 0$) must contain $[A_1^T B_1^T]^T$ and $[A_2^T B_2^T]^T$, but not $[A_3^T B_3^T]^T$.

8 Discussion

We have thus far highlighted the earliest stages in a mating operation, in which the moving part traverses the contact state lattice around the first critical point encountered in the trajectory. Traversal of this lattice begins in a region R_0 surrounding the initial target pose in free space. If there is more than one critical point, then the goal state in the first lattice must be present in the second lattice, where it serves as the initial state in the second sequence. The traversal of a contact state shared by two lattices constitutes a *global* transit between two critical points. To restrict the pose uncertainty resulting from such a transit, we must force the assembly to follow multiple-contact states, i.e., the crevices and corners of C-space [9][14]. If the contact state shared by adjacent lattices is 1-dimensional in $\Re^6$, then there will be virtually no uncertainty associated with the traversal of that state (except for uncertainty in the state's own location in $\Re^6$, which varies with

the toleranced part dimensions). When a new contact occurs, signalling the end of the global transition, the uncertainty region will be reduced to a single point in $\Re^6$, albeit lifted to $\Re^{t+6}$. The next lattice's traversal then commenses in a quite constrained uncertainty region.

The designer tools described in sections 2 – 6 have been implemented in the POPLOG environment running on a Sun 3 workstation. Surface descriptions of CSG-modelled assembly parts are obtained from a geometric modeller [3] and placed in POP-11 records, which serve as nodes in an adjacency graph of part facets [8]. Geometric data from the models parameterize linearized C-surface equations (2) and (3), to generate the contraint sets representing the primitive contacts' hyperplanar polytopes, as well as the convex components of *LEGAL*. A FORTRAN simplex procedure is employed to perform the feasibility tests mentioned in sections 5 and 6. The forward-projection algorithm described in section 6 was also implemented in POP-11, utilizing a 6–dimensional hyperrectangle boundary model to store and update the affine space descriptions of successively-projected uncertainty region facets. The algorithm relies heavily upon the Gram-Schmidt procedure, which performs the orthogonalization process in subroutine *perp*. Wrench constraints for jamming avoidance and contact maintenance in the chosen contact states are also generated, using the inequality formulae in section 7. Fine motion strategies for traversing contact state sequences, similar to those in figure 4a, have been designed interactively with this system.

As we described in sections 2 through 6, the process of designing a fine-motion trajectory involves (1) selecting a contact state sequence, (2) proposing command velocities (either directly or implicitly with target poses) for traversing the selected contact states, and (3) verifying that only the chosen states will be traversed in the presence of uncertainty. Alternatively, (1) or (2) may be reversed: the direct specification of command velocities determines not only the target poses, but also the contact states to be traversed. A target pose $\mathbf{dX}_i$ and a desired velocity $\mathbf{v}_i$ together define a ray in $\Re^6$. The first new primitive contact encountered along the ray becomes the next contact to establish. Moreover, the contact state formed by the additional contact is guarranteed to exist without overlap, since the new contact was the first encountered along the ray. In this alternative approach, the validation of legal contact states described in section 5 is unnecessary. With a view toward further automation, future research will explore the heuristic generation of command velocities (see, for example, Laugier [16]).

9 Conclusion

A methodology for augmenting a nominal assembly motion plan with a fine-motion strategy was introduced. Linearizations of C-space around critical points in the trajectory enabled us to use linear programming to synthesize *perturbed nominal trajectories* for traversing local, polytopic contact spaces in the presence of position, control, and model error.

10 References

1. Ambler, A.P., Popplestone, R.J. (1975), "Inferring the positions of bodies from spatial relationships", *Artificial Intelligence*, Vol. 6, No. 2, pp. 157-174.

2. Brooks, R.A. (1982), "Symbolic error analysis and robot planning", *International Journal of Robotics Research*, Vol. 1, No. 4 pp. 29-68.

3. Brown, C.M. (1982), "PADL2: a technical summary", *IEEE Computer Graphics Applications*, Vol. 2, No. 2, pp. 69-84.

4. Buckley, S,J. (1987), "Planning and teaching compliant motion strategies", PhD Thesis, M.I.T. Artificial Intelligence Laboratory, Cambridge, Mass. (also M.I.T. A.I. Lab T.R. 936)

5. Caine, M.E. (1985), "Chamferless assembly of rectangular parts in two and three dimensions", S.M. Thesis, Dept. of Mech. Eng., MIT.

6. Caine, M.E., Lozano-Pérez, T., Seering, W.P. (1989), "Assembly strategies for chamferless parts", *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 472-477.

7. Canny, J.F. (1989), "On computability of fine motion plans", *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 177-182.

8. Dakin, G., Liu, Y., Nair, S., Popplestone, R.J., Weiss, R. (1989), "Symmetry inference in planning assembly", *Proceedings of the IEEE International Conference on Robotics and Automation*, Vol. 3, pp. 1865-1868.

9. Donald, B.R. (1984), "Motion planning with six degrees of freedom", Technical Report AI-TR-791, Artificial Intelligence Laboratory, MIT.

10. Donald, B.R. (1986), "Robot motion planning with uncertainty in the geometric models of the robot and environment: a formal framework for error detection and recovery", *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1588-1593.

11. Erdmann, M. (1984), "On motion planning with uncertainty", Technical Report AI-TR-810, Artificial Intelligence Laboratory, MIT.

12. Erdmann, M. (1985), Using backprojections for fine motion planning with uncertainty", *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 549-554.

13. Fortini, E.T.(1967), *Dimensioning for Interchangeable Manufacture*, Industrial Press, New York.

14. Koutsou, A. (1986), "Parts mating by moving objects in contact", PhD Thesis, Dept. of Artificial Intelligence, Edinburgh University.

15. Latombe, J.C. (1989), "Motion planning with uncertainty: on the preimage backchaining approach", in *The Robotics Review*, (eds) O. Khatib, J. Craig, T. Lozano-Pérez, Cambridge, Mass.: MIT Press.

16. Laugier, C. (1989), "Planning fine motion stategies by reasoning in contact space", *Proceedings of the IEEE International Conference on Robotics and Automation*, pp.653-659.

17. Liu, Y. (1989), "Planning for assembly from solid models", *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 222-227.

18. Liu, Y. (1990), "Symmetry groups in robotic assembly planning", PhD Dissertation, COINS Dept., Univ. of Mass, Amherst, Mass. 01003.

19. Liu, Y., Popplestone, R.J. (1990), "Symmetry constraint inference in assembly planning: automatic assembly configuration specification", *Proceedings of the National Conference on Artificial Intelligence*, Boston Mass., July 29 - August 3, 1990.

20. Lozano-Pérez, T. (1976), "The design of a mechanical assembly system", Technical Report AI TR 397, AI. Lab, MIT.

21. Lozano-Pérez, T., Mason, M., Taylor, R.H. (1984), "Automatic synthesis of fine-motion strategies for robots", *International Journal of Robotics Research*, Vol. 3, No. 1, pp. 3-24.

22. Ohwovoriole, M.S., Hill, J.W., Roth, B.(1980), "On the theory of single and multiple insertions in industrial assemlies", *Proceedings of the 10th International Symposium on Industrial Robots*, Milan, Italy, March 1980, pp. 545-558.

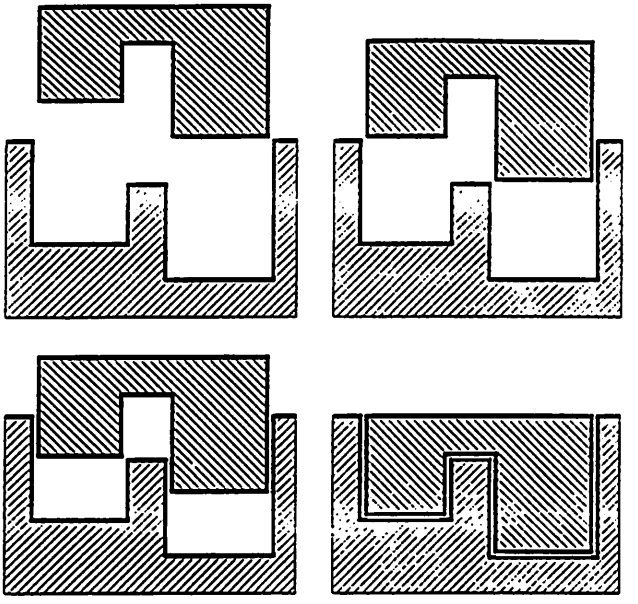
23. Popplestone, Ambler, A.P., Bellos, I. (1980), "An interpreter for a language describing assemblies", Artificial Intelligence, Vol. 14, No. 1, pp. 79-107.

24. Simunovic, S. (1975), "Force information in assembly processes", *Proceedings of the 5th International Symposium on Industrial Robotics*, pp. 415-427.

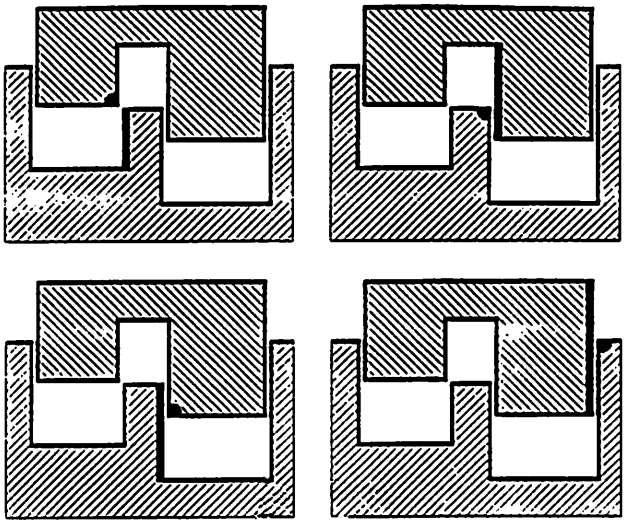
25. Taylor, R.H. (1976), "A synthesis of manipulator control programs from task-level specifications", Standford Univ., A.I. Lab., Memo AIM-282.

26. Whitney, D.E. (1982), "Quasi-static assembly of compliantly supported rigid parts", *Journal of Dynamic Systems, Measurement, and Control*, Vol. 104, pp. 64-77.

27. Whitney, D.E., De Fazio, T.L., Gustavson, R.E, Graves, S.C., Abell, T., Cooprider, C., Pappu, S. (1989), "Tools for strategic product design", *Preprints, Proceedings of the NSF Engineering Design Research Conference*, University of Massachusetts, Amherst, Mass., pp. 581-595.



(a)



(b)

Figure 1: (a) Critical points in a nominal trajectory. (b) Primitive contacts at a critical point.

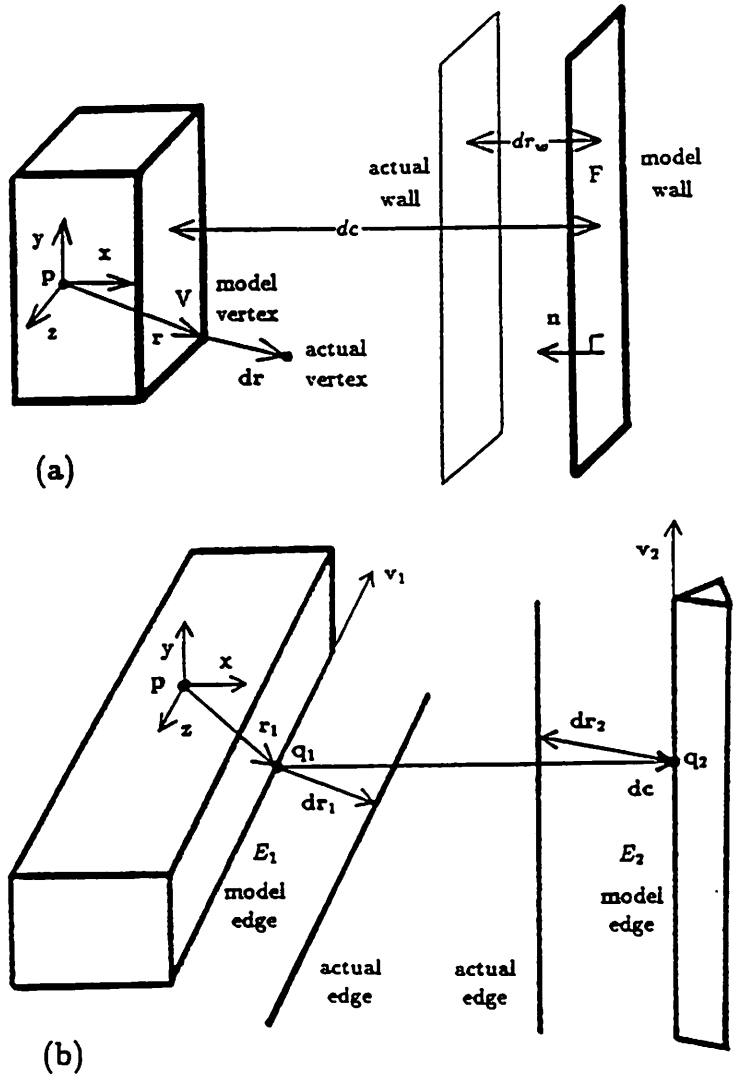


Figure 2: (a) A vertex-face contact. (b) An edge-edge contact.

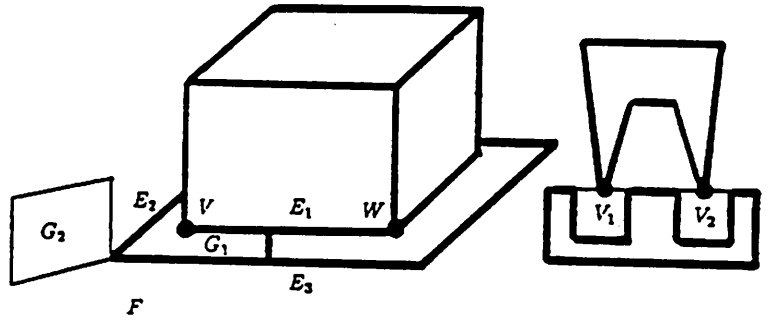


Figure 3: (a) A critical point in the nominal trajectory. (b) Rim vertices.

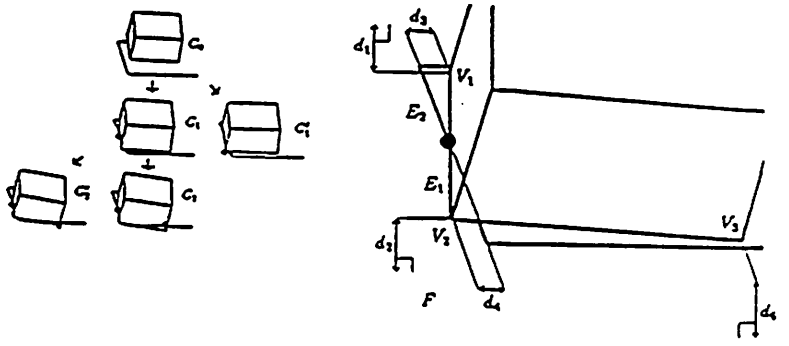


Figure 4: (a) Alternative sequences of contact states. (b) Specifying a target pose in a contact state.

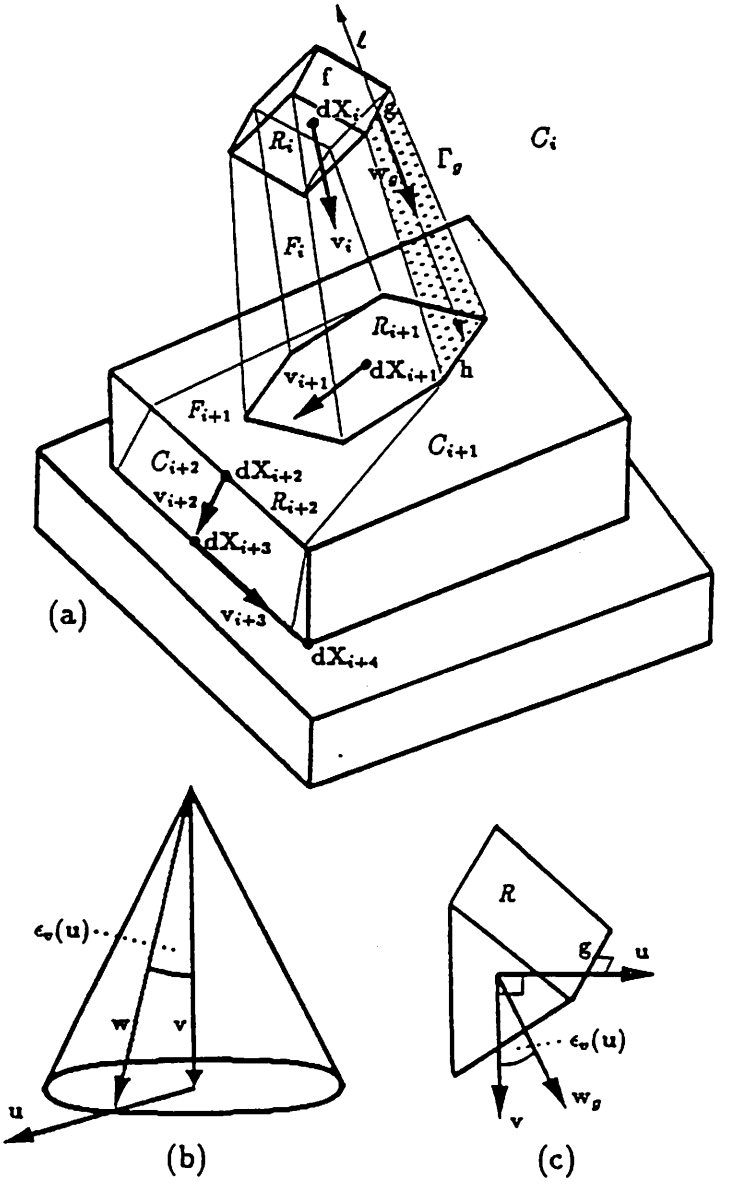


Figure 5: (a) Uncertainty analysis in C-space. (b) Control velocity error cone. (c) Deriving the motion of an uncertainty region facet.

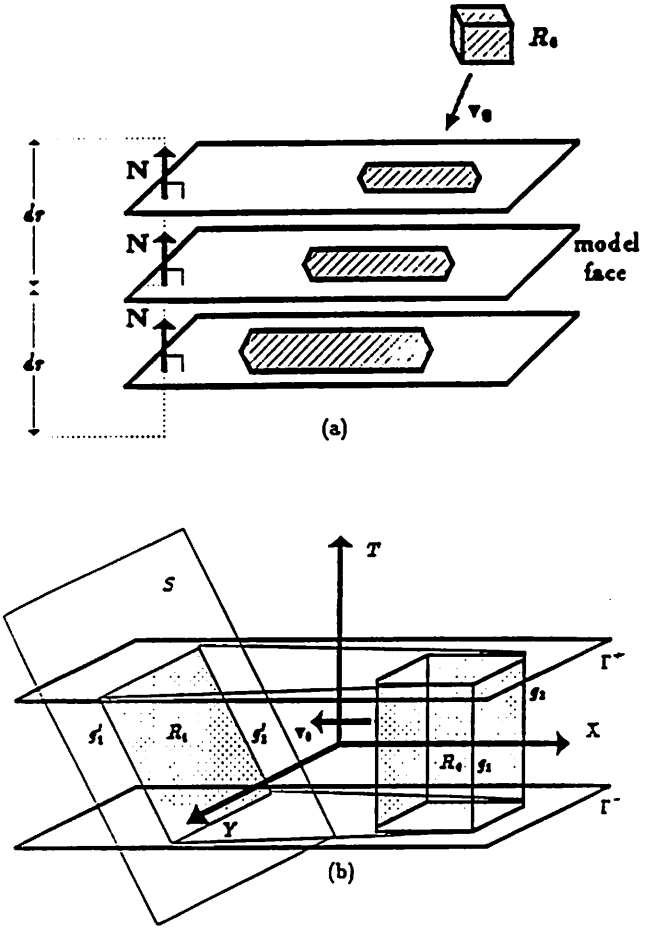


Figure 6: (a) Uncertainty region projected onto a toleranced surface. (b) A forward-projection in generalized C-space .

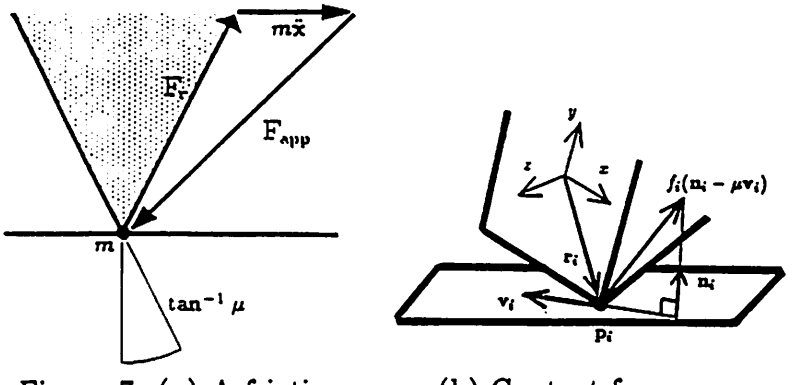


Figure 7: (a) A friction cone. (b) Contact force.

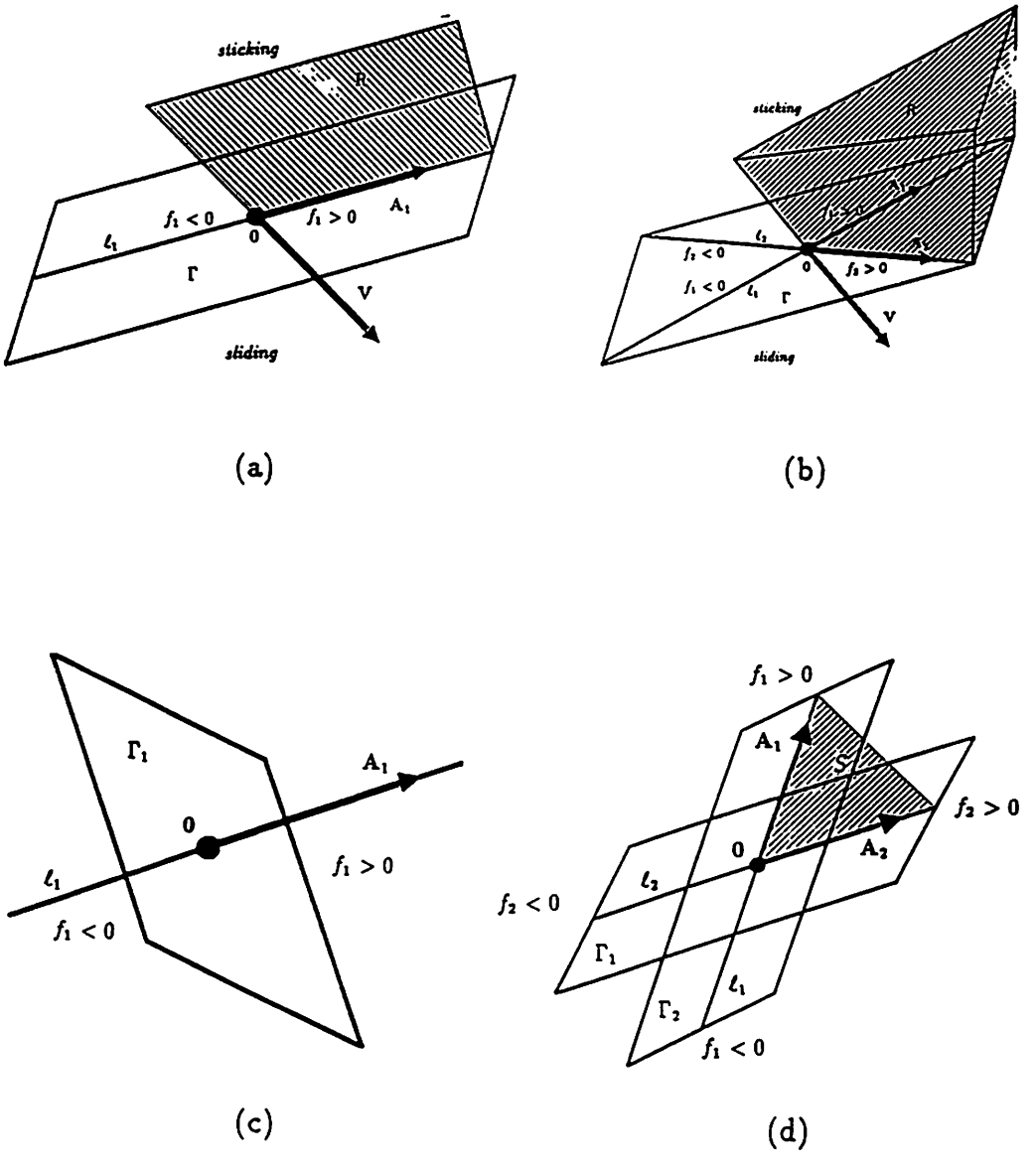


Figure 8: Jamming avoidance for 1-point contacts. (b) Jamming avoidance for 2-point contacts. (c) Maintaining a 1-point contact. (d) Maintaining a 2-point contact.