

# **Automatic Generation of Gripping Positions<sup>1</sup>**

**Jan D. Wolter<sup>2</sup>**  
**Richard A. Volz**  
**Anthony C. Woo**

**February 1984**

**CENTER FOR ROBOTICS AND INTEGRATED MANUFACTURING**

**Robot Systems Division**

**COLLEGE OF ENGINEERING**

**THE UNIVERSITY OF MICHIGAN**

**ANN ARBOR, MICHIGAN 48109**

---

<sup>1</sup>This work was supported by the Air Force Office of Scientific Research under contract F49620-82-C-0089, and in part by the Zimmer Foundation.

<sup>2</sup>The authors are with the Robot Systems Division of the Center for Robotics and Integrated Manufacturing, College of Engineering, University of Michigan.



### **Abstract**

This paper addresses the problem of automatically determining gripping positions for objects based upon geometrical knowledge of the object and its environment. Both the question of what constitutes a good grip and the efficiency of the computational algorithms are considered. Several new criteria for the stability of a grip are proposed, resilience to slippage and twisting in the gripper due to external forces and torques. Resilience to slippage is expressed in terms of friction effects of the surfaces involved, the shape of the contact between the gripper and object and the distance of the grip point from the center of mass of the object. Two new performance measures are introduced to reflect resilience to twisting. A general grip planning strategy is introduced which is both run-time efficient and robust. The problem is divided into three parts, one which depends only upon the geometry of the object to be gripped, one which takes into account a priori geometric constraints, and one which handles constraints unknown until run-time.



## 1. Introduction

Robot programming today is usually approached at one of two levels – manual training, or off-line procedural level programming. In a manually trained system it is very difficult to incorporate any decision making mechanisms. Thus such systems can only be used in rigidly defined applications where no adaptive or error handling capabilities are needed. Procedural level programming, on the other hand, allows decision making capabilities to be incorporated in the program, vastly increasing the versatility of the system but correspondingly bringing increased complexity. The programmer must specify explicitly each motion of the robot and verifying the correctness of robot programs becomes very difficult. Task level of programming, in which operations are specified in terms of their effects on objects in the work area, has been proposed to solve these problems. For example, instead of telling the robot to move through a point in space, to another point, and close the hand, one would simply instruct the robot to pick up a given part. To achieve a working task level system, task instruction must be translated into a sequence of specific actions by a *task planner*. Unfortunately, the development of good task planners is very difficult. Indeed, the most advanced task planning system proposed, AUTOPASS [1] the planning of grip positions on an object, and presents an effective method of determining these positions.

Grip position determination requires consideration of several basic issues: definition of what constitutes a good grip; specification of the geometries involved; knowledge of the environment; and computational methods. Typical conditions for a "good grip" include the *reachability*, [2]-[3] of the grip position by the robot without interference [4] between the robot and the object to be gripped, and the *stability*, [5]-[7] the nonmovability of the object being grasped under external forces. Asada [5] has given alternate definition of stability: that a grip is stable if when a small relative displacement occurs between the part and the hand a restoring force is generated to bring the part back to its original situation. Both Asada [5] and Paul [2] have also suggested that a good heuristic for maintaining stability is to grip the object near its center of mass.

The geometries of the objects to be grasped have been obtained both by visual input [8] and Computer Aided Design [2], [4], [6], [7]. Most authors, have considered only polyhedral

models and/or cylindrical models.

The determination of grasp positions cannot be accomplished based upon the object alone. Grasp positions depend upon other objects in the vicinity which might interfere with a particular grasp point as well. Several authors [9]-[12] have developed techniques for taking into account interfering objects in the vicinity of the initial object position. Lozano-Perez [10], [11] and Wingham [12] consider determination of a collision free path to an initial grasp configuration, and Lozano-Perez [10] also considers interference at the final position of the object after moving. Laugier and Pertin [4] point out that avoidance of grip positions which would cause interference with obstacles at either the initial or final positions can be easily incorporated into their algorithm.

Since the gripping problem is usually underconstrained, there are usually many acceptable grip positions. The calculations are often based upon generating a set of potential grips and then eliminating those that are thus not feasible or cause interferences. Distance from the center of mass and contact area are then sometimes used to pick a single grasp point [6], [13].

This paper addresses both the questions of what constitutes a good grip and the efficiency of computing grip positions. Several new criteria for the stability of a grip are proposed, resilience to slippage and twisting in the gripper due to external forces and torques. Resilience to slippage is expressed in terms of friction effects of the surfaces involved, the shape of the contact between the gripper and the object and the distance of the grip point from the center of mass of the object. Two new performance measures are introduced to reflect resilience to twisting. It is found that area is not directly a factor in the resilience to either slippage or twisting.

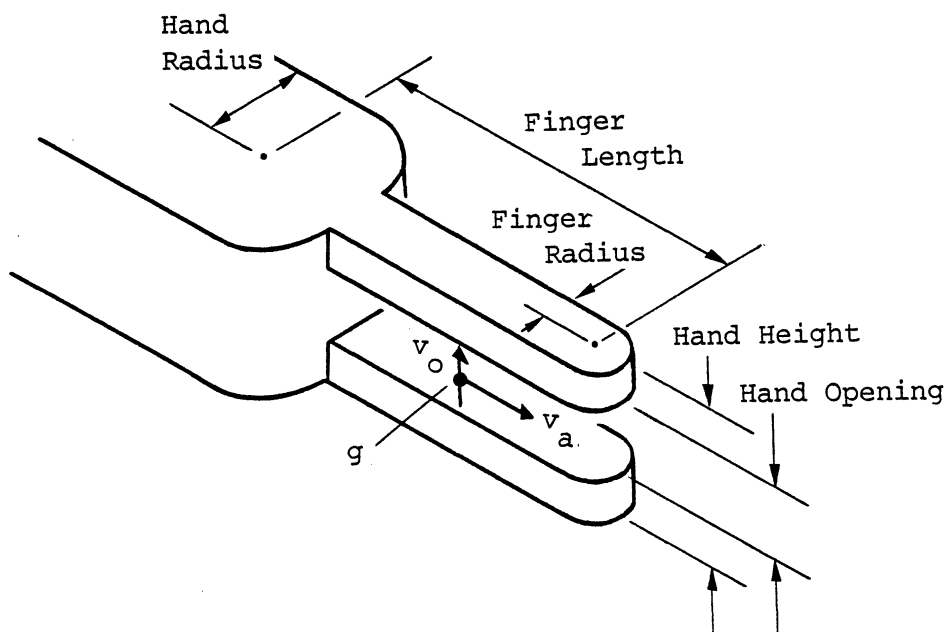
A general grip planning strategy is introduced which, though in some respects similar to the computational strategy described above, is both run-time efficient and robust in that it can be used in a broad range of applications. The problem is divided into three parts, one which depends only upon the geometry of the object to be gripped, one which takes into account a priori geometric constraints, and one which handles constraints unknown until run time.

The next section presents a formal statement of the problem. Section 3 discusses the problem and lays the motivation for the computational strategy, which is presented in Section 4. Section 5 describes the techniques for evaluating the stability of a grip and Section 6 presents an example of using grip determination system.

## 2. Problem Statement

Let  $P$  be a polyhedral part which is to be grasped and moved. Associated with  $P$  is an intrinsic coordinate system with respect to which points in  $P$  are specified. The location of  $P$  with respect to the world coordinate system of the robot is specified by a transformation relating this intrinsic coordinate system to the robot coordinate system. Two positions of the part are considered, the initial position, and the final position, represented by transformations  $p_i$  and  $p_f$  respectively.

Let  $G$  be a parallel finger gripper and hand of a robot, as shown in Figure 1. This model is designed to simplify obstacle evaluation, while still reflecting the morphology of a typical



**Figure 1** Hand model and parameters

parallel-jaw gripper. Specification of a grip position consists of a position vector,  $g$ , a unit approach (deproach) vector,  $v_a$ , parallel to the axis of the fingers and a unit orientation vector,  $v_o$ , perpendicular to the pads of the fingers. The triple  $(g, v_a, v_o)$  specifies the pose of the hand at the point where the gripper should be closed. It is convenient to express  $g$ ,  $v_a$  and  $v_o$  relative to the part intrinsic coordinate system. While  $P$  is in the grasp of the robot, this relationship will remain fixed, even though the hand and part may move.

In general, a grip pose cannot be determined on the basis of the part and gripper alone. Any obstacles which may be in the vicinity of  $P$  at either  $p_i$  or  $p_f$  must also be considered. Let  $O_j$ ,  $j=1, \dots, N$  be a set of polyhedral obstacles. In specifying a grip triple, it will be necessary to enforce a noninterference constraint between  $G$  and  $P$  and between  $G$  and the  $O_j$ ,  $j=1, \dots, N$  for both  $P$  at  $p_i$  and  $P$  at  $p_f$ .<sup>3</sup> To specify these constraints, it is necessary to consider the volume swept by  $G$  as it moves some distance  $L$  along the approach vector from the grip point. Thus, let,

$$\hat{G}(l) = \text{space occupied by } G \text{ when it is located at } g + l \cdot v_a \text{ and oriented along } v_o$$

Then, the problem can be stated as follows:

P1)  $G$  must remain fixed with respect to  $P$  during the motion from  $p_i$  to  $p_f$ ,

P2)  $\hat{G}(l) \cap P = \phi \quad \forall l \in [0, L]$

P3)  $\hat{G}(l) \cap O_j = \phi \quad \forall i=1, \dots, N \quad \forall l \in [0, L]$

where it is understood that this condition

must be applied twice, once for  $P$  at  $p_i$

and once for  $P$  at  $p_f$ .

P4)  $P$  will not twist (defined more precisely below) in the gripper as the gripper is closed on the part.

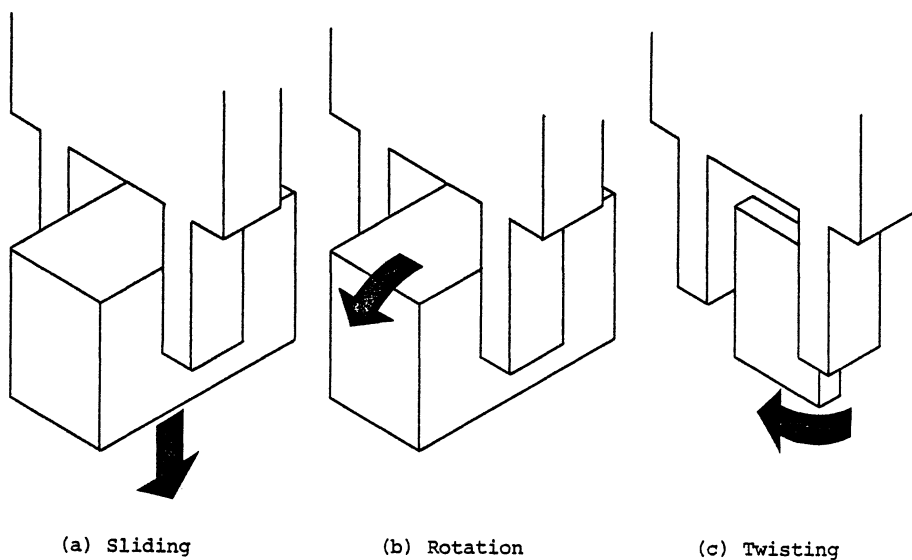
---

<sup>3</sup> Actually, one must consider the interferences of the  $O_j$ ,  $P$  and  $G$  for all points of  $P$  and along the trajectory from  $p_i$  to  $p_f$ . This, however, is relegated to the path planning problem and is not considered further here.



Figure 2 shows three kinds of motion which can occur between  $P$  and  $G$  due to gravitational and inertial forces, two kinds of slippage and twisting. One of the principle objectives of this paper is to develop quantifiable criteria which relate to the occurrence of these kinds of motion and can be used to test the satisfaction of condition P1. Conditions P2 and P3 simply indicate that there be no interference between the gripper and the hand or between the gripper and the obstacles.

Condition P4 also relates to twisting, but twisting due to a different cause. While P1 prohibits twisting due to inertial forces during motion, P4 refers to the occurrence of twisting while the grippers are being closed. Errors in the position or orientation of the gripper at the time the gripper are closed could result in hitting the part on one face before the opposite face is contacted, or contacting an edge of the part rather than a face. In turn this could cause the part to twist while the gripper is being closed. A second goal in the work present is to develop criteria which relates to twisting during grasping.



**Figure 2** Types of possible motion between gripper and part.

Computationally, the goal is to perform whatever run-time computations are required fast enough to allow their use in a real time system. Significant emphasis is placed on this goal and the structure of computational strategy reflects this in a major way.

### 3. Problem Analysis

#### 3.1. Off-line vs. On-line Computation

As indicated in P2 and P3, interference calculations must be performed to assure that a proposed grip causes neither interference between the gripper and the part nor interference between the gripper and any other objects in the vicinity of the initial and final positions. In a completely unstructured environment, all of the calculations for determining gripping positions are unknown until run-time and all processing must take place then. The manufacturing environment, however, is seldom completely unstructured, and one can use relevant knowledge about the environment to allow some of the computations to be performed off-line to improve run-time efficiency.

To provide motivation for the computation strategy to be used one can classify the structuring of the environment in a way that naturally leads to succession of strategies for determining grip positions with the amount of off-line computation dependent upon the amount of a priori knowledge. As an example the following table gives the amount of knowledge for some typical applications. In class C0, nothing is known in advance. The part description itself must be derived by a vision or some type of sophisticated sensor system and none of the needed computations can be performed off-line. Clearly, no advance planning is possible. In all of the other cases at least the geometry of the parts and gripper are known before run-time. One can conceptually, at least, perform the interference test between  $P$  and  $G$  off-line.

For class C1 the obstacle positions at the initial and final location are not known a priori. It is presumed, however, that an external sensor system does exist to identify the obstacles and determine their pose. Consequently, the interference checks between  $G$  and the  $O_j$ ,  $j=1, \dots, N$  must be done at run-time.

In class C2 it is assumed that the geometry and pose of some of the obstacles are known in advance. A common example of this would be the part lying in one of its stable positions on a table top (the table is the known obstacle) with other objects scattered about nearby. (The stable positions can be found from the geometric model by computing the convex hull of the part, and testing each face for stability [14].) Then the interference calculations between  $G$  and a subset of the  $O_j$ ,  $j=1, \dots, N$  might be performed off-line, thus reducing the amount of run-time computation. If there are no obstacles, other than the table, near the part (class C3), then the single best grip for each stable position of  $P$  can be chosen in advance. In class C4 the pose of  $P$  in space is known in advance, as in machine unloading. The exact move command to position the gripper can then be generated in advance.

These considerations motivate a computational strategy which is split between off-line and on-line computation and in which grip selection is done in four stages. First a set of grips is proposed and interference between  $G$  and  $P$  is evaluated. Second, interference between  $O_j$ ,  $j=1, \dots, N$  is evaluated. Third a single grip is selected and finally an exact move command is generated.

### 3.2. Slippage and Twisting Criteria

Ideally, if one had full knowledge of all physical parameters such as coefficient of friction, gripping force, and accelerations present, one could compute the forces and torques which would be necessary cause slippage of the object being grasped. Further, if one had full knowledge of the trajectories to be followed, one could compute the acceleration forces and torques to which the object would be subjected during motion. Based upon these, one could determine whether or not a proposed grip would result in slippage or twisting of the object being grasped.

Unfortunately, not all of the parameters which would be needed to operate in this ideal mode are readily calculable. Instead formulas are derived for related parameters which bear a monotonic relation to the force and torques required to cause slippage or twisting. These parameters can then be used as quality measures of a proposed grip, in which the higher rated grip positions are more resilient to slippage or twisting than lower rated positions. In particular, "quality"

Advance Knowledge in Robotics Applications					
Class	Off-line Knowledge Availability				Typical Application
	Part Geometries	Part Configurations	Local Obstacle	Part Pose	
C0	Unknown	Unknown	Unknown	Unknown	Unstructured
C1	Known	Unknown	Unknown	Unknown	Bin-Picking
C2	Known	Known	Partially Known	Known	Pick part from Cluttered Table
C3	Known	Known	Known	Unknown	Pick part from Clear Table
C4	Known	Known	Known	Known	Machine Unloading

Table 1

measures are developed for the following:

E1: A measure of the resilience of a grip to slippage of the object.

E2: A measure of the resilience to twisting of the object during motion of the hand with the part in its grasp.

E3: A measure of the resilience to twisting of the object during the closing of the gripper.

These evaluation criteria are then used in the grip determination strategy. Their detailed definition and evaluation mechanisms are discussed in Section 5.

#### 4. Strategy for Grip Determination

Grip determination is generally an underconstrained problem in that there are usually many grip positions which would be adequate; it is sufficient to find any one satisfying conditions P1-P4. This fact, together with the idea of being flexible in handling when the interference constraints P3 are imposed (off-line or on-line), leads to the strategy for determining the grip triple  $(g, v_a, v_o)$ . Rather than trying to directly obtain a unique  $(g, v_a, v_o)$  triple, a set of geometrically feasible triples satisfying P2 are computed. Since it would be desirable for this set to include grips which can be used for a large number of actual configurations of obstacles around  $p_i$  and  $p_f$ , this set should be reasonably dense on the part.

Next, each triple in the set is tested against the conditions P1-P4. The computations can be performed in two stages, one off-line and one on-line. The intent, of course, is to push as much of the computation to the off-line stage as possible. The computations associated with conditions P1, P2 and P4 can be performed off-line. Those associated with P3 depend on when the locations of the  $O_j$  become known; in general, they may be split between on-line and off-line. This strategy can be summarized as follows:

*Strategy I -- Grip Determination*

- S1: Determine a set of grip positions for the part (a method for list generation is described in the next section) satisfying condition P2.
- S2: Evaluate these feasible grips according to criteria E1 - E3 (the method for doing this is described in Section 5). Discard those falling below some set of threshold values and sort those remaining in rank order according to a combination of these criteria E1-E3.
- S3: Beginning with the highest ranked grip, apply the constraints P3 until a grip in the list is found which satisfies all constraints. Use this grip.

This strategy provides a flexible way of handling constraints and a technique to minimize run-time computation. For several classes of problems most of the calculations can be done off-line. Once a single grip has been chosen, it remains only to generate the motion commands to move the hand into position. The gross movement of the hand into the vicinity of the part is handled by a separate subsystem. The final approach to the part is made along the approach vector. The hand model used in intersection testing below ensures that this is possible.

#### **4.1. Grip List Generation**

The first step in the procedure for determining grip positions is the generation of a set of feasible grip positions (step S1 in strategy I). The initial grip list is generated using only knowledge of the geometry of the part and gripper. This information is extracted from a CAD database. Each object is described by its surface elements: faces, edges, and vertices. The grip list computation needs to be done only once for each part/gripper combination. First, pairs of

surface elements upon which to grip are selected. Next specific approach vectors are proposed for grips on each pair of surfaces. Each of these is then tested for collisions with the hand. In summary:

*Strategy II - - Grip List Generation*

- S1.1: Select pairs of surface elements which are nearly parallel, facing away from one another, separated by less than the opening of the hand, and opposite each other. Each such pair determines an orientation vector.
- S1.2: For each pair of selected surface elements generate a finite set of possible approach vectors.
- S1.3: Perform test P2 for points on each proposed approach vector. Discard those that fail. For each of the feasible vectors choose a possible grip point  $g$  on the vector.

Each of these steps is discussed in greater detail in the sections below.

#### **4.1.1. Surface Element Pair Selection**

Though surface elements which may be paired for potential gripping includes faces, groups of edges or vertices, only gripping on pairs of surfaces is considered here. Each face record in the model contains pointers to all adjacent edges, and the parameters of it's surface equation,

$$Ax + By + Cz + D = 0 \quad (1)$$

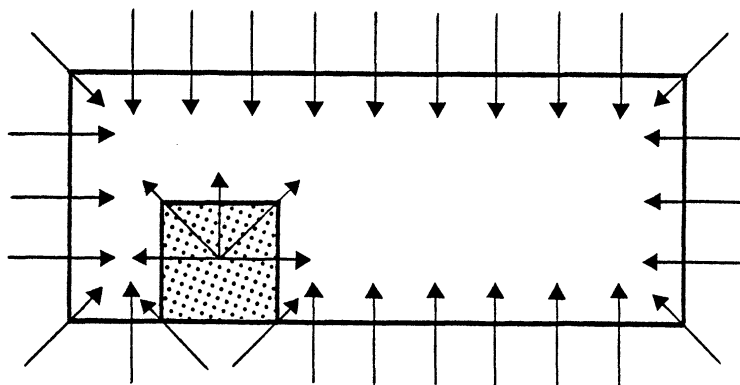
The vector  $(A,B,C)$  has unit length and points outward from the face. Each pair of faces is passed through a series of filters which eliminate geometrically infeasible pairs.

- (1) To ensure that the two faces be parallel within an angle of  $\theta_\epsilon$  of the sum of their unit normal vectors must be less than  $2\sin(\theta_\epsilon/2)$ .
- (2) The sum of the  $D$ s in the face equations yields the distance between the faces. The pair is rejected if the distance is greater than the finger separation.
- (3) To check that two faces are have nonzero overlap, the outlines of the two faces are projected on top of each other, and intersected. If the area of the intersection, is less than  $\epsilon_A$  the faces are rejected.

#### 4.1.2. Approach Vectors

Specification of a pair of surface elements for gripping determines an orientation angle  $\nu_o$ . Next, specific grip positions and approach vectors on these surfaces must be chosen. In order to ensure that a grip can be found in as many run-time situations as possible, a diverse set of grips should be proposed at this stage. In [4] this is done by computing the centroid of the grippable surface, and then finding a discrete set of directions from this point which are free of obstacles. These directions are proposed as approaches for the hand. This has several disadvantages. It constrains all grips to pass through the centroid of the surface, which in general has no special virtue. Grips near the ends of a long side tend to cross the edge at a very shallow angle. Also, large sections of the gripping surface which lie in the shadow of an obstacle may be ignored.

In order to propose a more diverse set of approach vectors, an algorithm driven on the boundaries of the grippable surface is used. Along each side of the surface, several approach vectors spaced uniformly along, and perpendicular to, the edge are proposed. At each corner of the surface, one approach vector is proposed bisecting the angle. (See Figure 3).



**Figure 3** Distribution of proposed grip positions on feasible surface

It is helpful to always propose an odd number of approaches along each edge. This ensures that there will always be one in the center of each edge, which is often an optimal position. Obstacles are not considered at this stage.

#### **4.1.3. Evaluating Obstacles on the Part**

Once a set of grips have been proposed, each must be tested to see which ones can be achieved without the gripper intersecting with the part (condition P2). The process can be simplified by first reducing the 3D intersection problem to a 2D intersection problem for each pair of grippable surfaces. The process proceeds in two stages, first projecting pertinent parts of the object and gripper on a plane parallel to the proposed gripping surfaces, and then performance of two dimensional interference checking.

##### **4.1.3.1. Projection of Obstacles onto Gripping Surface**

To check for interference between the gripper and the object for a potential grip position, three parts of the gripper must be considered, two fingers and the base of the hand (see Figure 1). First, the portion of the part which lies between the top and bottom planes of the hand is projected onto a surface parallel to the potential grip surfaces. A similar operation is performed for each finger. In the case of the fingers, however, their movement perpendicular to the gripping surface is taken into account. Accordingly, the cutting planes are defined by the outside finger planes with the fingers open and by the inside finger planes with the fingers closed. The inside finger planes, then, lie on one of the potential gripping surface. The portion of the part between each pair of finger planes is then projected on a surface parallel to the potential gripping surface. This is illustrated in Figure 4. Note that these projections need only be done once per pair of potential gripping surfaces, regardless of how many grip positions have been proposed for the pair of surfaces.

The idea of using cutting planes and checking obstacles in 2D is, on the surface similar to that proposed by Laugier and Pertin, but the algorithm used here to find the obstacles is different. Laugier and Pertin use a series of recursive plane slices through the part, bounded by an



arbitrary parametrization. A simpler, more efficient exact algorithm is described here using only a single set of cutting planes.

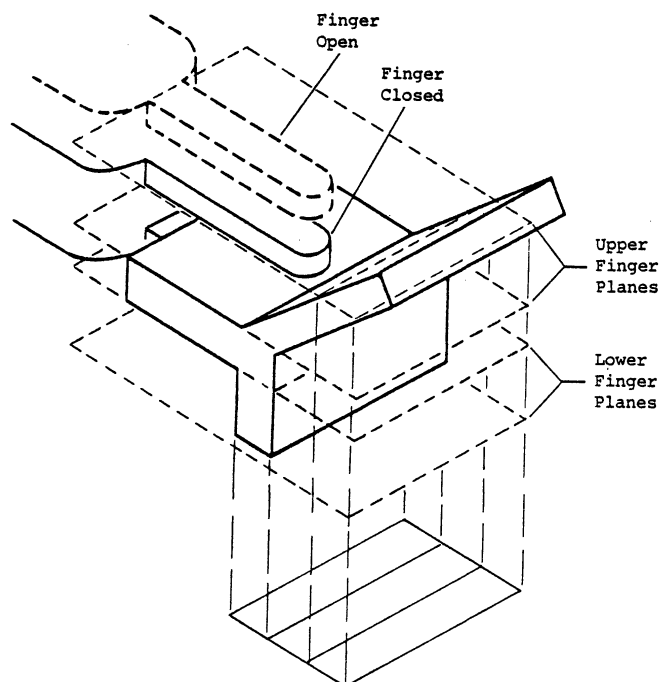
For each pair of cutting planes the algorithm recognizes three classes of line segments which are to be projected onto the grip surface. They are

- (1) all edges which lie entirely between the two planes,
- (2) for edges which cross either plane, the segments which lie between the two planes,
- (3) for faces which have any edges in class two, new segments connecting the intersection points between the edges and the planes.

When projected into 2D, these line segments form closed regions into which the projection of the hand of fingers (as appropriate) may not intrude.

#### 4.1.3.2. Intersection Testing

Once the projection has been completed, it is simple to check for obstacles along each of the approach vectors proposed for that pair of surfaces. The gripper model, when projected onto the



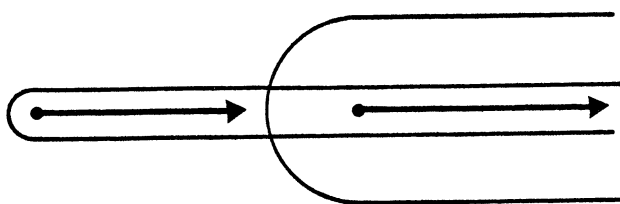
**Figure 4** Cutting planes along top and bottom of fingers determine portion of part projected on to gripping planes.

grip surface is just a pair of areas swept by sliding a pair of circles along rays corresponding to the spines of the hand and finger projections (see Figure 5). The hand ray is compared with the hand obstacle figure, and the finger ray is compared with the finger obstacle figure. The same sub-routine is used to evaluate both cases.

The intersections are found quickly. It is only necessary to locate the first point along the approach vector which is at the radius of the hand or finger circle (as appropriate) from some edge in the appropriate obstacle figure. The minimum of distance between the first hand obstacle and the first finger obstacle gives us the closest attainable point along the approach vector. Comparing this to the limits of the grippable surface found earlier tells us if the grip is possible.

## 5. Evaluation of Gripping Positions

Normally, a substantial number of feasible grips will be generated by the above procedure. Each is to be evaluated according to the criteria E1 - E3 stated earlier, slippage and twisting. Two distinct kinds of slippage are possible: translational (Figure 1a) and rotational (Figure 1b). Twisting (Figure 1c) is also shown.



**Figure 5** Projection of Hand and Fingers  
onto Gripping Planes.

### 5.1. Translational Slippage

The translational force required to cause slippage of a part in the hand depends only on the force applied by the fingers and the coefficient of static friction between the part and the gripper's fingers. The typical parallel finger gripper applies a constant force which is not under control of the robot programmer. Then, under the assumptions of constant force on the grip contact and constant coefficient of friction, the force needed to cause translational slippage is independent of the grip. Thus, resistance to translational slippage does not generally give rise to a useful criteria for rating grips.

### 5.2. Rotation Slippage

Rotational slippage depends on the torque applied to the part, the pressure of the gripper on the part, the shape of the contact between the gripper and the part, and the coefficient of friction of the surface. To calculate the torque required to rotate a part between the fingers of the gripper let the contact surface be as shown in Figure 6, and assume that the applied torque will cause the object to rotate about a point  $C$ . Treat  $C$  as the origin of a polar coordinate system and let the pressure on the contact surface be  $P(r, \theta)$  at a point  $(r, \theta)$ . The force required to cause sliding of an elemental area would be

$$\delta f = \mu_s P(r, \theta) r d\theta dr \quad (2)$$

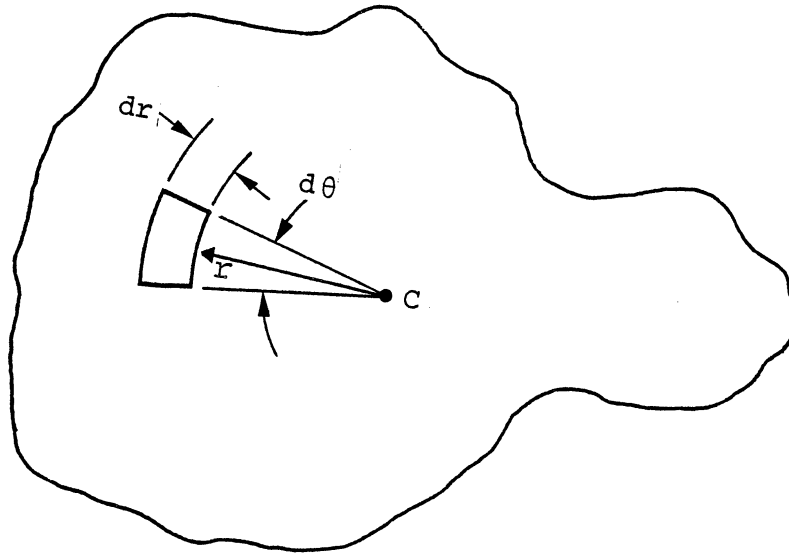
where  $\mu_s$  is the coefficient of static friction. The corresponding torque to cause slippage for that elemental area is

$$\delta \tau = \mu_s P(r, \theta) r^2 d\theta dr \quad (3)$$

The rotational torque required to cause slippage is found by integrating  $\delta \tau$  over the contact area.

$$\tau_s = \mu_s \int_0^{2\pi} \int_0^\infty P(r, \theta) r^2 \phi(r, \theta) dr d\theta \quad (4)$$

where  $\phi(r, \theta)$  is a characteristic function describing the shape of the area of contact. This function is one for all points on the contact surface, and zero for points not on the contact surface:



**Figure 6** Contact Area and Elemental Section

$$\phi(r, \theta) = \begin{cases} 1 & \text{if } (r, \theta) \text{ is a point of contact} \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

It is assumed that the pressure applied to contact surface is constant. In this case the torque required to cause rotation can be written as

$$\begin{aligned} \tau_s &= P\mu_s \int_0^{2\pi} \int_0^{\infty} r^2 \phi(r, \theta) dr d\theta \\ &= \frac{N\mu_s \int_0^{2\pi} \int_0^{\infty} r^2 \phi(r, \theta) dr d\theta}{\int_0^{2\pi} \int_0^{\infty} r \phi(r, \theta) dr d\theta} \end{aligned} \quad (6)$$

where  $N$  is the normal force applied by the finger.

For example, if the contact surface is a ring of outer radius one and of inner radius  $r \in [0, 1]$  then the torque required is

$$\tau_s = \frac{2}{3} n\mu_s \frac{1 + r + r^2}{1 + r} \quad (7)$$

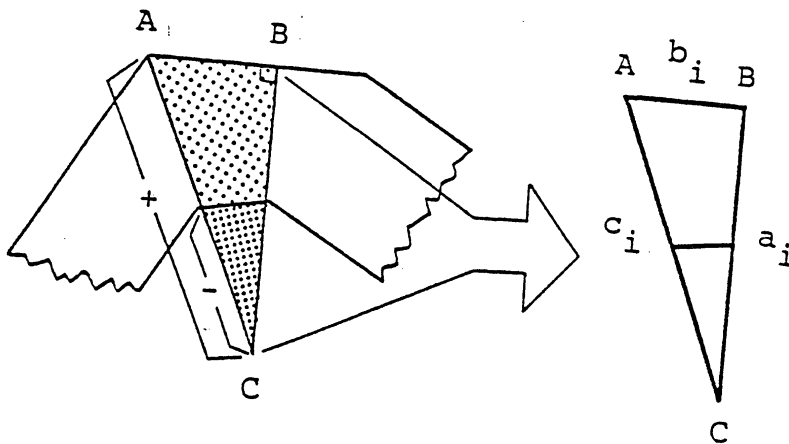
As  $r$  increases toward one, the area diminishes to zero while  $\tau_s$  increases to  $N\mu_s$ . This illustrates the fact that contact area is not necessarily a good measure of resilience to slippage.

To evaluate this exactly for any polygonal area of contact, the polygon is first decomposed into right triangles each with one acute vertex on the center of rotation. For each edge of the polygon, two triangles are found. Both have one side on the perpendicular to the edge, through the center of rotation. The other sides are the edges from intersection between the perpendicular and the two endpoints. The hypotenuses are the segments connecting the center of rotation to the two endpoints of the edge. One or both of these triangles may be negative. This evaluation can be done in linear time, and always gives rise to  $N_E$  triangles where  $N_E$  is the number of edges in the polygonal contact area.

The integral (6) has an exact solution for a right triangle rotated around an acute vertex. If  $a_i$  is the side adjoining the center of rotation,  $b_i$  is the other side, and  $c_i$  is the hypotenuse (see Figure 7) solving the integral for the  $i^{\text{th}}$  triangle gives:

$$\tau_i = P\mu_s \left[ \frac{2}{3} a_i b_i c_i + \frac{1}{6} \ln \left( \frac{b_i + c_i}{a_i} \right) \right] \quad (8)$$

Let  $\text{sign}(i)$  be a function which is 1 if the  $i^{\text{th}}$  triangle is positive, and -1 if the  $i^{\text{th}}$  triangle is negative. Then the total torque required to rotate the polygon is:



**Figure 7** The triangle ABC is one of the elemental triangles used in the evaluation of eq.(6). The high density subtriangle is not part of the contact area and would be evaluated as a separate negative triangle.

$$\tau_s = P \mu_s \sum_{i=1}^{\Delta} \text{sign}(i) \left( \frac{2}{3} a_i b_i c_i + \frac{1}{6} \ln \left( \frac{b_i + c_i}{a_i} \right) \right) \quad (9)$$

where  $\Delta$  is the number of triangles. Since normal force, not pressure, is constant between grips, it simplifies comparison between grips if pressure is replaced with normal force over area. This gives the final form of the expression:

$$\tau_s = N \mu_s \frac{\sum_{i=1}^{\Delta} \text{sign}(i) \left( \frac{2}{3} a_i b_i c_i + \frac{1}{6} \ln \left( \frac{b_i + c_i}{a_i} \right) \right)}{\sum_{i=1}^{\Delta} \text{sign}(i) \frac{1}{2} a_i b_i} \quad (10)$$

Now, consider the gripper under acceleration. Let  $F_{a \max}$  be the maximum acceleration applied during the motion of the arm holding the object and assume that it is applied through the center of mass. It would be desirable to have

$$\tau_s > F_{a \max} \cdot D \quad (11)$$

where  $D$  is the distance of  $C$  from the center of mass. Or

$$\frac{\tau_s}{D} > F_{a \max} \quad (12)$$

Since the acceleration forces to be applied are not readily calculable,  $\tau_s / D$  is treated as a quality measure which should be large as possible.

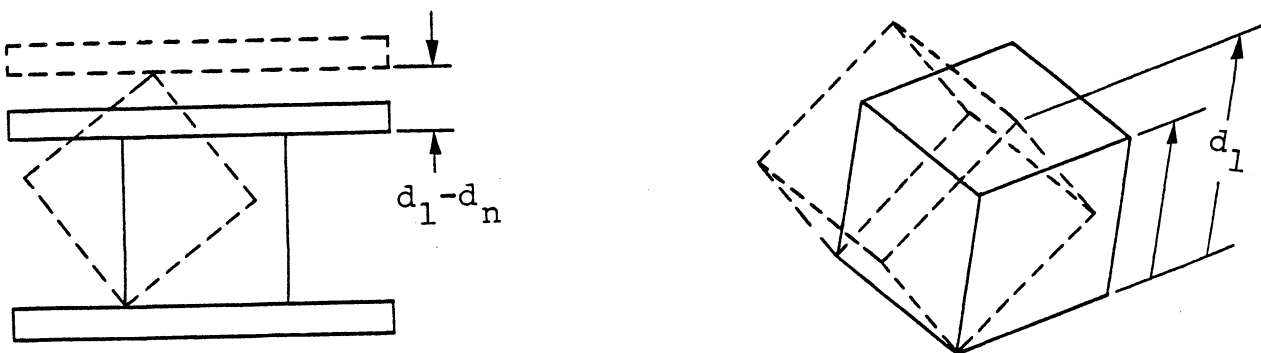
Two issues remain in the treatment of rotational slippage: both contact surfaces of the gripper must be considered; and the location of the point  $C$  must be specified. Including both contact surfaces of the gripper simply means that  $\tau_s$  is composed of two components  $\tau_s = \tau_{s1} + \tau_{s2}$  where  $\tau_{s1}$  and  $\tau_{s2}$  are each obtained by equation (10) for their respective contact surfaces. If the two contact surfaces are not of the same shape, opposite each other and in the correct orientation, the pressure distribution will not be uniform as assumed in equation (11). In this case, equation (11) corresponds to using the average pressure. The location of  $C$  is also not readily determined. It appears both in  $D$  and in  $\tau_s$ . When the center of mass is located substantially away from the gripper,  $C$  is approximately at the centroid of the two contact surfaces, and this is used in our calculations.

### 5.2.1. Twisting

Either acceleration torque or a misalignment of the gripper during approach to the gripper surfaces could result in the part twisting in the gripper. Two measures of resilience to twisting are considered. First, when acceleration torques cause a part to twist in the gripper, the fingers must be forced apart. One relevant measure is the distance by which the fingers must be separated for the part to twist out of position. Figure 8 shows how a grip on a part may become unstable when twisted slightly. To find how much it must be twisted, the convex hulls of the 2D contact areas of each finger with the surface of the part are computed. Let the set of edges in these convex hulls be  $H$ . For each edge  $e_i$  of one hull, find the most distant vertex on the other hull. This is how far the hand would have to open if tipped over that edge. Let this distance be  $d_i$ . The minimum of these distances is then found

$$d_t = \min_{e_i \in H} [d_i] \quad (13)$$

and the normal finger separation distance  $d_n$  is subtracted from  $d_t$ .



**Figure 8** Possible instability of a part can occur through twisting about one of its edges

$$\Delta d = d_i - d_n \quad (14)$$

$\Delta d$  is the additional distance the hand would have to open to allow the part to twist into another position. and is used as a measure of resilience to twisting during motion of the hand and part.

Secondly, errors in the location of the hand or the part at the time the gripper is closed may cause the part to twist in the gripper. The angle error which would results in a part to tipping over an edge becomes another measure. Let  $\alpha_i$  be the angle to tip over edge  $e_i$ . Then

$$\alpha = \min [\alpha_i] \quad (15)$$

$$e_i \in H$$

is the measure of the resilience to uncertainty in the relative position and orientation of the part and gripper during the closing of the gripper.

## 6. Examples

As an example, the part shown in Figure 9 was considered for two different hands used on robots in our laboratory. The part model has 44 faces, 132 edges and 88 vertices.

Steps S1 and S2 of Strategy I are first performed off-line. Table 2 shows the time required for this computation. The result is a list of possible grips which cause no interferences between the hand and the object together with values for the rating functions.

The geometry of the hand 1 allows more potential grips to be reached, hence the larger number of grips in this case. The geometry of the hand 2 led to a larger number of obstacle calculation; hence the larger cpu time.

Hand	No. of Grips Found	CPU Time (VAX 11/780)
Hand 1	113	2.0 sec
Hand 2	62	3.2 sec

**Table 2**

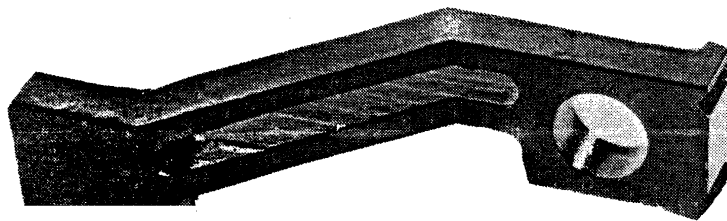


An issue which has not yet been resolved is a determination of the relative importance of the gripping criteria derived above. This is a matter which requires considerable additional study. Initial investigations are being performed with functions of the form:

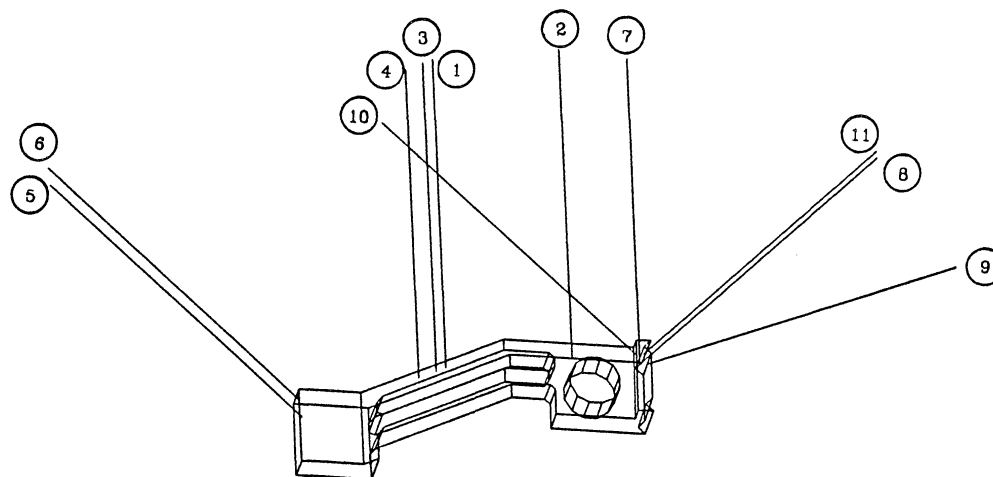
$$\frac{k_1 \Delta d + k_2 \tau_s / \mu_s N}{D + \epsilon} + k_3 \alpha \quad (16)$$

This form applies weightings  $k_1$ ,  $k_2$  and  $k_3$  to the three criteria.  $D + \epsilon$  is used rather than  $D$  in the denominator to keep the expression finite as  $D$  approaches zero. Since it appears that resilience to twisting during motion is also related to the distance from the centroid, the ratio of  $\Delta d$  to  $D + \epsilon$  is used instead of  $\Delta d$  alone. At present, values for  $k_1$ ,  $k_2$ ,  $k_3$ , and  $\epsilon$  are being chosen empirically.

Figure 10 shows one stable position for the object and the approach vectors proposed for gripping by hand 2 for  $k_1 = 9$ ,  $k_2 = 1$ ,  $k_3 = 0$  and  $\epsilon = 0.1$ . These are numbered according to the values produced by the rating functions, with number 1 being the most desirable grip. The



**Figure 9** Example part



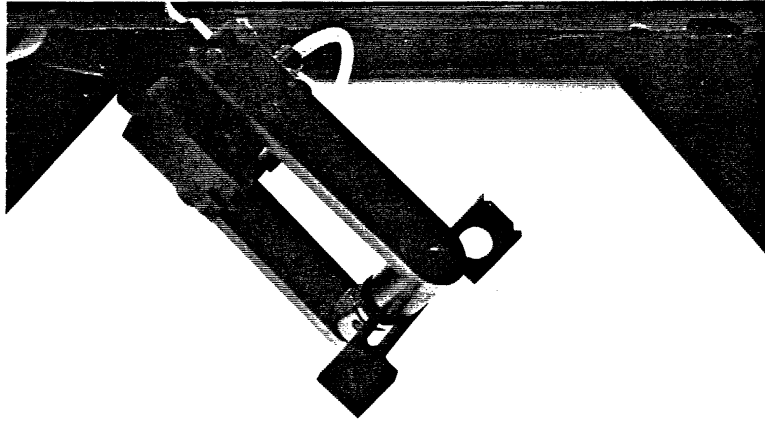
**Figure 10** Approach vectors for possible grip positions are shown with the rank ordering of the eleven most desirable grips.

ratings for the grips shown vary by a factor of five. The top eleven grips are identified; these are approximately within a factor of two of each other, with 1, 2 and 3 being significantly higher than the rest.

This gripping strategy has been incorporated into small robot cell [15] consisting of an Intel iAPX 432 multiprocessing computer system with several attached processors for controlling the vision and robot subsystems, and an ASEA rb6 robot. The system is totally CAD driven, with both the training of the vision system and grip determination derived from CAD models of the parts and obstacles. Figure 11 shows the part of figure 9 in the grasp of the robot of this cell.

## 7. Summary and Conclusions

This paper has described criteria for evaluating the quality of a prospective grip position which take into account friction effects, torsion on the proposed grip, the likelihood of twisting of the object in the grip and uncertainty in angle of approach to the prospective grip position. Also, a computationally flexible strategy for determining gripping positions was presented. First, a set



**Figure 11** The part in the robot hand

of feasible grips are determined and evaluated according to the criteria developed. Next intersection calculations between the hand, the object and the surroundings are performed, and the highest ranking position in the set which does not cause interference is selected. The strategy allows some of the computations to be shifted between on-line computation and off-line computation according to the amount of a priori information available.

There remain some unsettled issues. First, the relative importance of the evaluation criteria are not well understood. There is no direct inclusion of dynamics in the algorithm. If the acceleration forces and torques could be included properly, the quality criteria could be transformed into threshold conditions which, if satisfied, would ensure that the grip would not result in object slippage. Uncertainties in robot approach or object position are only partially taken into account; by E3. One might address the uncertainty problem by "growing" the size of the gripper in proportion to the uncertainties involved before performing the feasibility and interference calculations, but this has not been done yet, and further work is needed. Also, uniform pressure distributions were assumed during gripping; the effects of violating that assumption need further study. Finally, very little has been done with grippers configurations other than the

parallel gripper. As more general grippers come into use, much more work will be needed here.

## 8. References

- [1] L. Lieberman, "Autopass: a very high level programming language for mechanical assembler systems," RC 5599, No. 24205, T. J. Watson Research Center, IBM, Yorktown Heights, NY, Aug. 1975.
- [2] R. P. Paul, "Modelling, trajectory calculation, and servoing of a computer controlled arm," AIM 177, Artificial Intelligence Lab, Stanford Univ., Nov. 1972.
- [3] R. H. Taylor, "The synthesis of manipulator control programs from task-level specifications," AIM-282, Artificial Intelligence Lab, Stanford Univ., July 1976.
- [4] C. Laugier and J. Pertin, *Automatic grasping: a case study in accessibility analysis*. Barga, Italy: NATO - Advanced Studies Inst. on Robotics & Artificial Intelligence, June 1983.
- [5] H. Asada, *Studies in prehension and handling by robot hands with elastic fingers (Ph.D dissertation)*. Univ. of Kyoto, 1979.
- [6] J. Wolter, T. C. Woo, and R. A. Volz, "Gripping position for 3D objects," *Proc. of the 1982 Meeting of the Industrial Applications Soc.*, pp. 1309-1314, Oct. 1982.
- [7] C. Laugier, "A program for automatic grasping of objects with a robot arm," *Proc. of the 11th Int. Symp. on Industrial Robots*, pp. 287-294, Oct. 1981.
- [8] J. M. Brady, "Parts description and acquisition using vision," *Proc. SPIE*, May 1982.
- [9] P. Brou, *Implementation of high-level commands for robots (M.S. thesis)*. Dept. of Electrical Engrg. and Computer Science, MIT, Dec. 1980.
- [10] T. Lozano-Perez, "The design of a mechanical assembly system," AI TR 397, Artificial Intelligence Lab, MIT, 1976.
- [11] T. Lozano-Perez, "Automatic planning of manipulator transfer movements," *IEEE Trans. on Systems, Man, and Cybernetics*, vol. SMC-11, no. 10, Oct. 1981.
- [12] P. H. Winston, in *Artificial Intelligence*. Reading, Mass: Addison-Wesley Publishing Co., 1977.
- [13] M. Brady, J. M. Hollerbach, T. L. Johnson, T. Lozano-Perez, and M. T. Mason, in *Robot Motion: Planning and Control*, Brady, et al., Ed. Cambridge, MA and London, England: MIT Press, May 1983.
- [14] M. A. Wesley, T. Lozano-Perez, L. I. Lieberman, M. A. Lavin, D. D. Grossman, "A geometric modeling system for automated mechanical assembly," *IBM J. Res. Development*, vol. 24, no. 1, Jan. 1980.
- [15] R. A. Volz, A. C. Woo, J. D. Wolter, T. N. Mudge, J. L. Turney, and D. A. Gal, *CAD, Robot programming and Ada*. Barga, Italy: NATO - Advanced Studies Inst. on Robotics & Artificial Intelligence, June 1983.