Planning for Modular and Hybrid Fixtures

Aaron S. Wallack *   John F. Canny †

Computer Science Division

University of California

Berkeley, CA 94720

Abstract

Fixturing is a fundamental problem in mechanical assembly. Usually, two and a half dimensional objects can be fixtured in many different ways using a fixture vice, especially if pegs of different radii are available. We present an algorithm which enumerates all force closure fixture vice configurations and corresponding object poses. Automatic fixture design algorithms are essential for planning because optimal fixturing selection for multiple operations requires examining all of the valid configurations. The algorithm runs in $O(A)$ time, where A is the number of configurations which simultaneously contact the object.

1 Introduction

The task of immobilizing a workpiece via mechanical devices, commonly called fixturing or workholding, is an essential problem in manufacturing. Machining fixtures must handle very large forces (20KN), whereas assembly fixtures handle smaller forces (50N). Fixture apparatus design is more a craft than a science. Without geometric analysis, a fixturing expert system is capable only of describing "types" of fixturing components, not the positions of the fixtures and the object. This paper details a methodological, geometric, fixturing design algorithm.

As a first step towards designing an analytic fixture planning system, we analyze a nontrivial task: the task of enumerating all configurations for immobilizing a particular object using a fixture vice, a device commonly used in woodworking. A fixture vice consists of modular fixture elements (pegs) placed on fixture tables which are mounted on jaws of a vice, as shown in Figure 1. The fixture vice possesses the minimum number of degrees of freedom necessary (one) to deal with workpiece variations. Theoretically,

it can immobilize any generic two and a half dimensional object. It could also be used as an *adaptable* gripper. The fixture vice is based on three mechanical
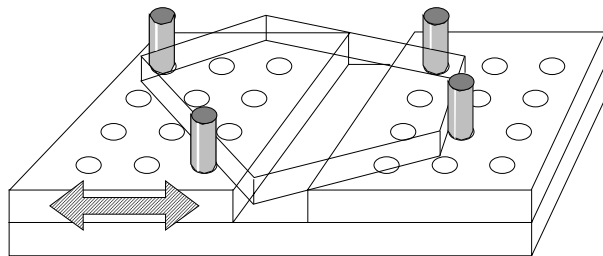

Figure 1: A fixture vice consists of two fixture table jaws capable of translating in x.

devices: pegs, a fixture table, and a vice. It combines simplicity and efficacy. Vice contacts can only occur at vertices, but fixture vice contacts can occur anywhere on an object. Fixture vices can hold objects without crushing corners, and can immobilize objects using internal holes. When using a fixture vice, one need only roughly identify the pose and verify that specified edges of the object contact specified pegs. This is because four generic edges can simultaneously contact four pegs in at most four different ways.

One advantage of using fixture vices is reducing changeover time by reusing part or all of the fixture for sequential operations. Usually, many different fixture configurations are capable of fixturing an object, especially if pegs of different radii are available. Since optimal selection requires examining all configurations, efficient, complete algorithms are essential to realize potential changeover time savings. In this paper, we present a polynomial time, complete algorithm for computing fixture configurations for immobilizing two and a half dimensional polyhedral objects. The algorithm works with the object's shadow and a two-dimensional representation of the fixture vice (refer Figure 2).

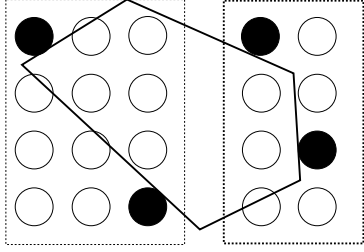Figure 2: A two-dimensional view of the object and fixture vice in Figure 1.

1.1 Related Work

This research stems from work in Reduced Intricacy Sensing and Control (RISC) robotics, which attempts to combine simple, modular hardware with intelligent software [3]. Hazen and Wright presented a thorough overview on research in automated fixturing components, techniques, planning, and execution [7] which suggested that most of the planning research focuses on expert systems. Asada and By developed the Automatically Reconfigurable Fixturing (ARF) system, which automatically synthesizes and constructs workholdings consisting of modular fixtures[1]. Markus et al. described an interactive expert system for designing workholdings composed of *towers*[10]. Mishra proved lower bounds on the number of toe clamps necessary to immobilize a rectilinear polyhedral object [11]. Brost and Goldberg described a complete algorithm for fixturing two and a half dimensional polyhedral objects using a fixture plate and a side clamp [2]. Force closure has been discussed in the grasping and robotics literature. Markenscoff et al. gave finite lower bounds on the number of fingers required to immobilize a two or three dimensional object with or without friction [9], as well as an efficient algorithm for computing finger placements on polygonal objects which minimize the maximum force necessary to counter any unit force through the center of mass [8]. Mishra et al. [12] used Steinitz's theorem to prove a lower bound on the number of fingers necessary for immobilization. Nguyen constructed independent regions of contact on polygonal and polyhedral objects over which all contacts produced force closure [13]. Ferrari and Canny [6] introduced two *quality criteria* for scoring grasp configurations. Faverjon and Ponce extended Nguyen's work on independent regions to curved two-dimensional objects [5].

1.2 Notation

- O refers to a two and a half dimensional polyhedral object.

- $\vec{E}$ refers to a quartet of *jaw-unspecified* edge segments.

- $\vec{\mathcal{E}}_i$ refers to the ith quartet of *jaw-specified* edge segments of object O, edge segments combined with fixture jaws they contact. $\mathcal{E}_{i,k}$ refers to the kth edge segment of the quartet $\vec{\mathcal{E}}_i$.

- $\vec{\mathcal{F}}_{i,j}$ refers to the jth quartet of peg positions contacting edge segments $\vec{\mathcal{E}}_i$. $\mathcal{F}_{i,j,k}$ refers to the kth fixturing position of the set of fixturing positions $\vec{\mathcal{F}}_{i,j}$.

- λ_{row} and λ_{column} refer to the spacing between the rows and columns respectively on the modular jaws (refer Figure 3).
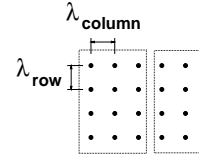

Figure 3: λ_{row} and λ_{column} refer to the spacing between the rows and columns respectively on the modular jaws.

- Bracketed expressions refer to continuous interval ranges of values: e.g., $< \theta_{consistent} >$ refers to the range of orientations consistent with the variable $\theta_{consistent}$. Ranges can be collected in sets: e.g., $\{< \chi_{consistent} >\}$ refers a the set of continuous boundary arcs. The $\oplus$ operator describes range addition: $< a, b > \oplus < c, d > = < a + c, b + d >$.

1.3 Overview

The configuration of a fixture vice describes the peg positions and radii. In this algorithm, we also use the term *configuration* to specify the object's pose, the peg positions $\vec{\mathcal{F}}$, and the jaw separation distance σ; the term *force closure* describes the capability of resisting arbitrary forces and torques. We assume frictionless point contacts between the object and the pegs. The algorithm runs in $O(A)$ time where A is the number of different sets of peg positions $\{\vec{\mathcal{F}}\}$ simultaneously contacting quartets of edges; there can be $O(n^4((\frac{\delta \rho}{\lambda_{row}\lambda_{column}})^2 \frac{\delta^2}{\lambda_{row}\lambda_{column}}))$ such peg configurations, where n is the number of edges, δ is the maximum distance between points in O, ρ is the maximum edge distance, and λ_{row} and λ_{column} are the row and column spacings respectively.

The algorithm generates all peg configurations simultaneously contacting each set of edge segments. For each set of edge segments, the problem is separated into enumerating combinations of peg positions

which simultaneously contact the edge segments, and computing the contacts between these components in order to verify force closure. Throughout this report, fixture configurations are described by peg positions $\vec{\mathcal{F}}$ and the corresponding edges $\vec{\mathcal{E}}$.

Algorithm:

1. Enumerate all jaw-specified edge segment quartets (combinations of four edge segments such that any edge segment can appear more than once) $\{\vec{E_a}, \vec{E_b}, \ldots\}$ of an object O capable of generating force closure .

2. For each quartet of jaw-unspecified edge segments $\vec{E_a}$, enumerate all different combinations of intended jaw contacts. Without loss of generality, only seven different situations need to be considered: four where three edges contact the left jaw, and three where pairs of edges contact both jaws.

3. For each edge segment quartet $\vec{\mathcal{E}}_i$, compute peg configurations $\{\vec{\mathcal{F}}_{i,1}, \vec{\mathcal{F}}_{i,2}, \ldots\}$ simultaneously contacting $\vec{\mathcal{E}}_i$ (refer Figure 4).
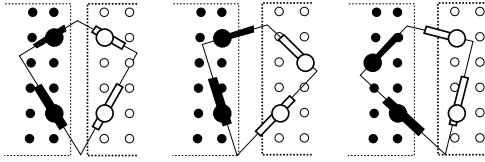

Figure 4: Different peg configurations $\vec{\mathcal{F}}_1$, $\vec{\mathcal{F}}_2$, $\vec{\mathcal{F}}_3$ simultaneously contacting edge segments $\vec{\mathcal{E}}$.

4. Compute the contact points between edge segments $\vec{\mathcal{E}}_i$ and pegs $\vec{\mathcal{F}}_{i,j}$, and verify force closure.

The bounds are achieved by assuming that the first peg is placed at the origin of the left jaw because without loss of generality, translated copies of configurations are redundant. The second peg on the left jaw must be inside an annulus around the origin $(\frac{\delta\rho}{\lambda_{row}\lambda_{column}})$, the peg on the right jaw must be inside a circle centered at the origin $(\frac{\delta^2}{\lambda_{row}\lambda_{column}})$, and the final peg must be inside an annulus of another peg on that jaw $(\frac{\delta\rho}{\lambda_{row}\lambda_{column}})$.

In section two, we present theoretical background and geometrical framework for both the enumeration algorithm and pose determination technique. In section three, we outline an algorithm which enumerates all possible sets of peg positions: $\{\vec{\mathcal{F}}_{i,1}, \vec{\mathcal{F}}_{i,2}, \ldots\}$ contacting a particular set of edges $\vec{\mathcal{E}}_i$: this set is constructed incrementally– first, generating all possible positions for the first peg, and then generating all possible positions for the second peg for each first peg

position, and so on. In section four, we describe a pose determination technique for computing the object's pose satisfying the condition that the edge segments $\vec{\mathcal{E}}_i$ contact the pegs $\vec{\mathcal{F}}_{i,j}$ mounted on translating jaws; thus verifying contact and grading the quality of the fixture configurations. We conclude by highlighting the results and advantages of this technique.

2 Theoretical Background

Theoretical background is introduced in this section.

2.1 Observations

We make two observations: cylindrical or flatted pegs (fixture elements) can be considered point contacts by suitably transforming the corresponding edge (refer Figure 5), and there are only two generic modes of simultaneous contact: Type I- two edges of the object contact pegs on each each jaw, and Type II- three of the object's edges contact pegs on a single jaw (refer Figure 6).
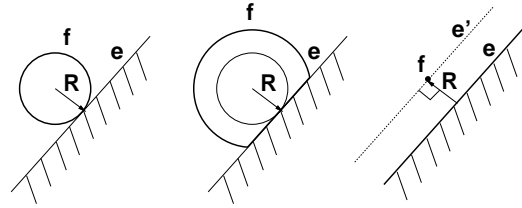

Figure 5: Cylindrical or flatted pegs, can be considered point contacts by suitably shifting the corresponding edge.
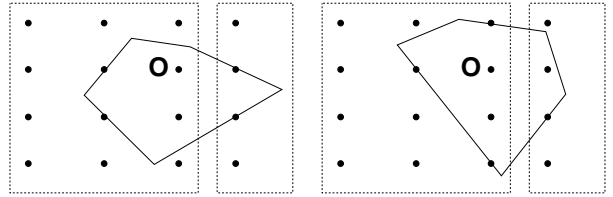

Figure 6: Type I (left): two edges of the object contact pegs on each jaw. Type II (right): three edges of the object contact pegs on a single jaw.

Claim 1 *At least one endpoint of all extremal length chords between any two edges coincides with a vertex of one of those edges.*

Claim 2 *The orientations of vectors between points on two edges form a continuous interval range, whose boundaries result from pairwise combinations of vertices of those edges.*

2.2 Four Degrees of Freedom are Necessary to Fixture Two and a Half Dimensional Objects

Generic fixturing techniques for two and a half dimensional objects require at least four degrees of freedom, and this is shown by a dimension counting argument: force closure requires four simultaneous contacts; satisfying four constraints generically requires four-degree-of-freedom systems, i.e., the fixture vice (x, y, θ: object, σ: fixture vice).

2.3 Paramaterization for Maintaining Contact Between Two Lines and Two Points

Let us parameterize the contact positions between two pegs and two edges as the angular position of the extended intersection of the edges on a circle. This parameterization is based upon the geometric property that the interior angle between a point on a circle's boundary and a circular arc remains constant. Let P_χ be the intersection (refer Figure 9) of the two lines (in a consistent frame), C be the circle including the two contact points $\mathcal{F}_{i,j,1}$, $\mathcal{F}_{i,j,2}$ and P_χ. Let A be an arc defined by the two points. The interior angle between $P\chi$ and A is equal to a only on when P_χ is on the circle's boundary (refer Figure 7). This suggests the parameterization: $\chi = \angle(P_\chi - center(C))$, which is used to compute the regions crossed by other edges (refer sections 3.5- 3.6). The method breaks down when the two lines are parallel; in those cases, there is another parameterization: the object's orientation is one of two orientations, and its position can translate parallel to those lines.
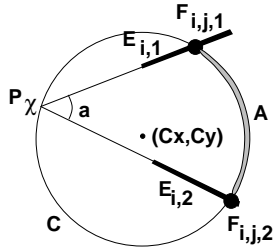

Figure 7: Parameterizing the object's pose by the position of the extended intersection P_χ on the circle's boundary maintains contact between $\mathcal{E}_{i,1}$, $\mathcal{E}_{i,2}$ and $\mathcal{F}_{i,j,1}$, $\mathcal{F}_{i,j,2}$.

3 Fixture Configurations Contacting a Set of Edges

In this section, we describe the peg configuration algorithm and subroutines.

3.1 Algorithm Outline

The algorithm efficiently enumerates all peg configurations $\{\vec{\mathcal{F}}_{i,1}, \vec{\mathcal{F}}_{i,2}, \ldots\}$ simultaneously contacting

a set of edge segments $\vec{\mathcal{E}}_i$. It exploits geometrical constraints between the peg positions and the edge segments and is outlined below:

1. Compute $< \theta_{consistent} >$, the orientations of the object O, and enumerate the possible positions of the second peg $\mathcal{F}_{i,j,2}$ assuming $\mathcal{F}_{i,j,1}$ is at the left jaw origin (refer sections 3.2, 3.3).

2. Parameterize the object's pose by the extended intersection, χ of $\mathcal{E}_{i,1}$, $\mathcal{E}_{i,2}$ (refer section 2.3), and compute $\{< \chi_{consistent} >\}$, the consistent interval ranges of χ (refer section 3.4).

3. Compute all positions of remaining pegs with respect to the left jaw by computing the discrete fixture rows crossed by $\mathcal{E}_{i,3}$ and $\mathcal{E}_{i,4}$ (refer section 3.5), and then computing the range of x coordinates crossed on each row (refer section 3.6).

4. Depending upon the type (I or II) of contacts, compute the peg positions on their intended jaws from these x ranges (refer sections 3.7, 3.8).

3.2 $< \theta_{consistent} >$: Object Orientations

The object's orientation θ must conform to the fact that, without loss of generality, an edge which contacts a peg on the left jaw cannot lie entirely to the right of an edge contacting a peg on the right jaw; the term $\theta_{consistent}$ refers to the orientations resulting from intersecting this constraint over all pairs of opposite jaw edges.

3.3 Positions of the Second Peg $\mathcal{F}_{i,j,2}$.

The second peg is constrained in two ways: the distance between $\mathcal{F}_{i,j,1}$ and $\mathcal{F}_{i,j,2}$ must agree with the distance between two points on $\mathcal{E}_{i,1}$ and $\mathcal{E}_{i,2}$, and the orientation between $\mathcal{F}_{i,j,1}$ to $\mathcal{F}_{i,j,2}$ is constrained to lie within the range $< \omega_{consistent} >$ (refer equation (1)). These constraints imply that $\mathcal{F}_{i,j,2}$ must lie within a wedge of an annulus defined by $< \omega_{consistent} >$, $minRadius_i$, and $maxRadius_i$ (refer equations (1)- (3) and Figure 8).

$$min\theta_i = \min_{v_1 \in \mathcal{E}_{i,1}, v_2 \in \mathcal{E}_{i,2}} \angle(v_1 - v_2)$$

$$max\theta_i = \max_{v_1 \in \mathcal{E}_{i,1}, v_2 \in \mathcal{E}_{i,2}} \angle(v_1 - v_2)$$

$$< \omega_{consistent} > = < \theta_{consistent} > \oplus < min\theta_i, max\theta_i > \quad (1)$$

$$minRadius_i = \min_{v_1 \in \mathcal{E}_{i,1}, v_2 \in \mathcal{E}_{i,2}} |v_1 - v_2| \quad (2)$$

$$maxRadius_i = \max_{v_1 \in \mathcal{E}_{i,1}, v_2 \in \mathcal{E}_{i,2}} |v_1 - v_2| \quad (3)$$

3.4 $\{< \chi_{consistent} >\}$: Positions of Extended Intersections

The position of the extended intersection P_χ must satisfy the constraints described by $\theta_{consistent}$ and
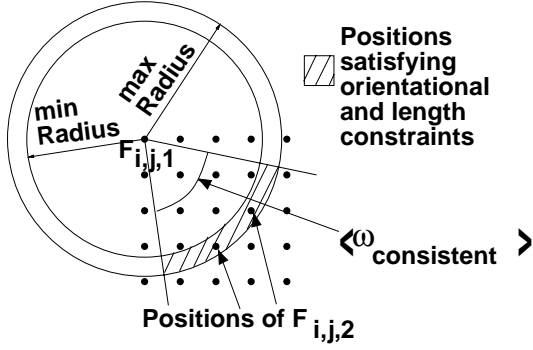
Figure 8: $\mathcal{F}_{i,j,2}$ is inside a wedge of the annulus defined by $minRadius_i$, $maxRadius_i$, and $< \omega_{consistent} >$.

$\omega_{consistent}$. The term $\chi_{consistent}$ refers to the extended intersections satisfying both of these criteria.

The extended intersection parameterization breaks down when the extended edges contact the pegs but the actual edge segments do not. This condition is tested by determining the minimum and maximum distances from the extended intersection to both edges $\mathcal{E}_{i,1}$, $\mathcal{E}_{i,2}$ (ζ_i refers to the minimum distance, and ξ_i refers to the maximum distance). χ must lie on an arc between the annulus described by $\mathcal{F}_{i,j,1}$, ζ_1, and ξ_1 as well as the annulus for $\mathcal{E}_{i,2}$ and $\mathcal{F}_{i,2}$.
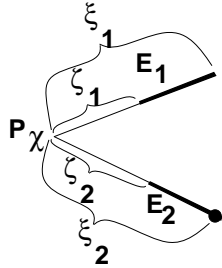

Figure 9: P_χ refers to the extended intersection of the first edge $\mathcal{E}_{i,1}$ and the second edge $\mathcal{E}_{i,2}$.

3.5 Fixture Rows $\{\mathcal{R}_{i,k,l}\}$ Crossed by Edge $\mathcal{E}_{i,k}$

The fixture rows crossed by the edge segment $\mathcal{E}_{i,k}$ are computed by determining the continuous range of y coordinates covered by $\mathcal{E}_{i,k}$, while maintaining contact between the first two edges and two pegs (refer figure 11). The extremal y coordinates in this range result from vertices $v_{i,k,m}$ of $\mathcal{E}_{i,k}$. Over each continuous range of $< \chi_{consistent} >_n$, the y coordinate range for each vertex is found by testing the extremal orientations of χ, and also orientations of locally extremal y values. Locally extremal orientations are found by substituting $u = \tan(\frac{\chi}{4})$ and solving for u in $\frac{dY(u)}{du} = 0$ (refer Figure 12, equations (4), (5))). In equations (4) and (5), γ refers to the interior angle between
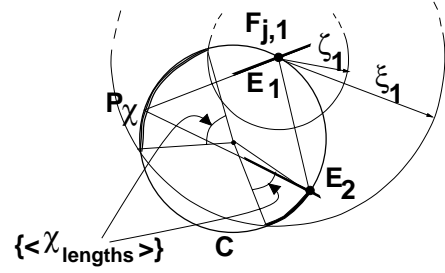

Figure 10: The minimum and maximum edge lengths ζ_i, ξ_i constrain $\{< \chi_{consistent} >\}$, the orientation corresponding to the position of the extended intersection P_χ along the circle's boundary.
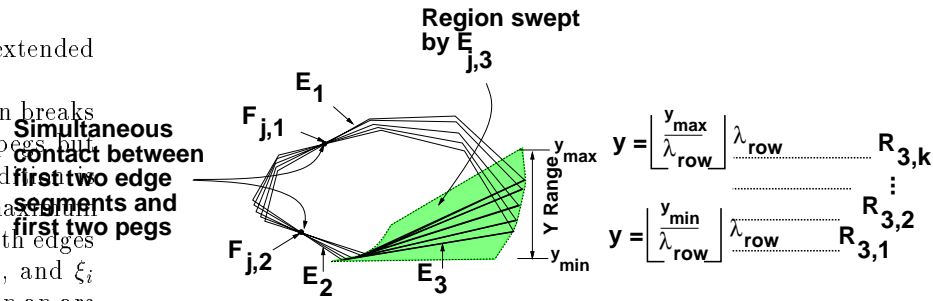

Figure 11: The fixture rows intersecting the region swept by $\mathcal{E}_{i,k}$ while maintaining contact between $\mathcal{E}_{i,1}$, $\mathcal{E}_{i,2}$ and $\mathcal{F}_{i,j,1}$, $\mathcal{F}_{i,j,2}$.

$\overline{P_\chi \, \mathcal{E}_{i,2}}$ and $\overline{P_\chi \, v_{i,k,m}}$; S refers to the distance between P_χ and v; β refers to the orientation of $\mathcal{E}_{i,2}$'s contact point ($\mathcal{F}_{i,j,2}$) with respect to the center of the parameterizing circle C; R refers to the radius of the parameterizing circle C. Given the set of discrete fixture rows crossed by $\mathcal{E}_{i,k}$, the next step is to determine the x coordinate ranges along each row.
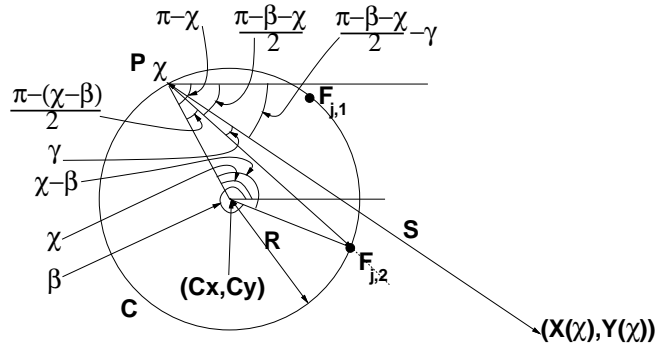

Figure 12: The position $X(\chi), Y(\chi)$ of a point making interior angle γ with the line from P_χ to the second edge $\mathcal{E}_{i,2}$ and distance S

$$Y(\chi) = C_y + R \sin 2\chi - S \sin(\frac{\pi - \beta}{2} - \gamma - \chi))$$

$$\tau = \frac{\pi - \beta}{2} - \gamma$$

$$Y(u) = C_y + \frac{1}{(1+u^2)^2}(R4u(1-u^2)$$
$$-S(1+u^2)((1-u^2)\sin(\tau) - 2u\cos(\tau))) \quad (4)$$

$$\frac{dY(u)}{du} = \frac{4}{(1+u^2)^3}(R + \cos(\tau)S + S\sin(\tau)u - 6Ru^2$$
$$+ S\sin(\tau)u^3 + Ru^4 - \frac{1}{2}\cos(\tau)Su^4) \quad (5)$$

3.6 $< J^{\mathcal{R}} >_{\mathcal{E}}$: X **Coordinate Range for Peg** $\mathcal{F}_{i,j,k}$ **for** $\mathcal{E}_{i,k}$ **On Fixture Row** $\mathcal{R}_{i,k,l}$

In this section we compute $< J^{\mathcal{R}} >$, the x co-ordinate range along a fixture row crossed by an edge segment. The first step involves computing $\{< \chi_{consistent}^{\mathcal{R}_{i,k,l}} >\}$ which combines $\{< \chi_{consistent} >\}$ with the constraint that $\mathcal{E}_{i,k}$ crosses the fixture row $\mathcal{R}_{i,k,l}$. $\{< \chi_{consistent}^{\mathcal{R}_{i,k,l}} >\}$ is computed by sorting all of the χ orientations s.t. $Y(\chi) = \mathcal{R}.y$ and the extremal orientations of $\{< \chi_{consistent} >\}$, and testing the intermediary ranges.

Figure 13 shows the model used in computing the x coordinate of the extended intersection, termed $J(\chi)$ (refer equation 6), between an edge segment and a horizontal fixture row as a function of χ ($u = \tan(\frac{\chi}{4})$). In figure 13, ϕ refers to the interior angle between $\overline{P_\chi \mathcal{E}_{i,k}^{normal}}$ and $\overline{P_\chi \mathcal{E}_{i,2}}$, and Q refers to the minimum distance between P_χ and $\mathcal{E}_{i,k}$. The range $< J^{\mathcal{R}} >$ is computed by checking the boundaries of $< \chi_{consistent}^{\mathcal{R}} >_i$ as well as orientations corresponding to locally extremal x coordinates, i. e. $\frac{dJ(u)}{du} = 0$ (refer equation (7)). There are at most two orientations u with locally extremal x since the numerator of $\frac{dJ(u)}{du}$ is a quadratic polynomial.
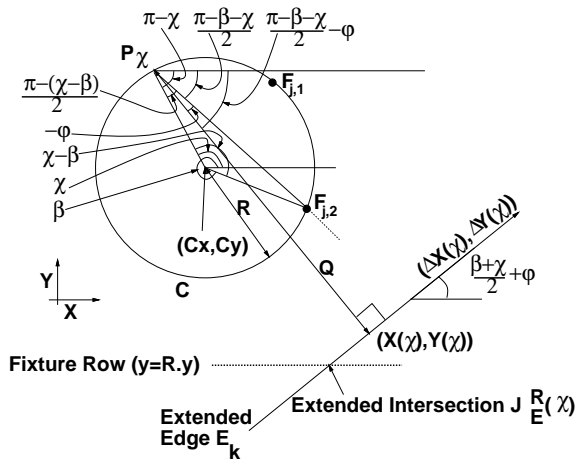

Figure 13: The position of the intersection $J(\chi)$ of the fixture row and the extended edge.

$$(X,Y)(\chi) = (C_x + R\cos 2u + Q\cos(\frac{\pi - \beta}{2} - \phi - u),$$

$$C_y + R\sin 2u - Q\sin(\frac{\pi - \beta}{2} - \phi - u))$$

$$\mu = \frac{\pi - \beta}{2} - \phi$$

$$X(u) = C_x + \frac{1}{(1+u^2)^2}(R((1-u^2)^2 - 4u^2)$$
$$+ Q(1+u^2)(\cos(\rho)(1-u^2) + \sin(\rho)2u))$$

$$Y(u) = C_y + \frac{1}{(1+u^2)^2}(R4u(1-u^2)$$
$$- Q(1+u^2)((1-u^2)\sin(\mu) - 2u\cos(\mu)))$$

$$\Delta X(u) = \frac{1}{1+u^2}(\cos(\frac{\beta}{2} + \phi)(1-u^2) - \sin(\frac{\beta}{2} + \phi)2u)$$

$$\Delta Y(u) = \frac{1}{1+u^2}(\cos(\frac{\beta}{2} + \phi)2u + \sin(\frac{\beta}{2} + \phi)(1-u^2))$$

$$J(u) = X(u) + (H - Y(u))\frac{\Delta X(u)}{\Delta Y(u)} \quad (6)$$

$$\frac{dJ(u)}{du} = \frac{2}{Z(u)}((u\cos(\mu) - 1)2Q +$$
$$(C_y - H + Q\sin(\mu) - \cos(2\mu)R)(1+u^2)) \quad (7)$$

3.7 Peg Positions $\mathcal{F}_{i,j,3}$, $\mathcal{F}_{i,j,4}$ (Type I Contact)

The main idea is that, even though the interval ranges were computed with respect to the left jaw, the differential between the x coordinate ranges for $\mathcal{E}_{i,3}$ and $\mathcal{E}_{i,4}$ remains valid with respect to the right jaw. For each combination of rows, $\mathcal{R}_{i,3,l}$, $\mathcal{R}_{i,4,l'}$, valid peg positions correspond to discrete values ($k\lambda_{column}(k \in \mathbb{I})$) in this differential range: $\{< J_{\mathcal{E}_{i,j,3}}^{\mathcal{R}_{i,3,l}} >\} \oplus \{- < J_{\mathcal{E}_{i,j,4}}^{\mathcal{R}_{i,4,l'}} >\}$ (refer Figure 14).

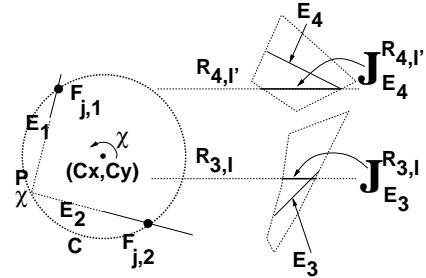Figure 14: $\{< J^{\mathcal{R}} >\}$ denotes the range of x coordinates of points on an edge intersecting a modular row.

3.8 Peg Positions $\mathcal{F}_{i,j,3}$, $\mathcal{F}_{i,j,4}$ (Type II Contact)

For Type II situations, $\mathcal{F}_{i,j,3}$ positions are on fixture rows $\mathcal{R}_{i,3,l}$ at discrete points ($k\lambda_{column}(k \in \mathbb{I})$) within $< J^{\mathcal{R}_{i,3,l}} >$. $\mathcal{F}_{i,j,4}$ positions on the right jaw are the leftmost positions on the fixture rows intersecting the remaining edge segment $\mathcal{E}_{i,4}$ when the object contacts the first three pegs (there are at most four such poses).

4 Computing Poses Such That Edge Segments $\vec{\mathcal{E}}_i$ Contact Pegs $\vec{\mathcal{F}}_{i,j}$

This section describes a technique for computing the object's poses such that edge segments $\vec{\mathcal{E}}_i$ simultaneously contact the pegs $\vec{\mathcal{F}}_{i,j}$ (refer Figure 15). The contact constraints are reformulated algebraically in order to compute the object's poses. The poses are computed using the relative displacements between contact points on the same fixture; there are two such relative displacements which constrain the placement of the part. Each relative displacement specifies a curve in (θ, y) configuration space where (θ, y) represents the object after a rigid two-dimensional transformation of rotation by θ and translation by y parallel to the y-axis; x is assumed to be zero since both jaws can translate freely along the x axis.

In both Type I and Type II situations, there are two independent relative displacements between contact points: in Type I situations, each jaw's pair of contacts supplies a relative displacement constraint, and in Type II situations, any two pairwise combination of contacts on the triple-contact jaw supply two relative displacement constraints. There are at most four poses for Type I contacts and two poses for Type II contacts. The configurations which satisfy
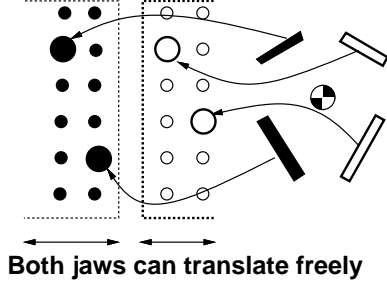

Both jaws can translate freely

Figure 15: Determine pose such that edge segments $\vec{\mathcal{E}}_i$ simultaneously contact pegs $\vec{\mathcal{F}}_{i,j}$ on fixture jaws which freely translate along the x axis.

the relative displacement constraints are described by algebraic curves. These two curves, called constant extended intersection difference (C) curves (refer section 4.3), are defined in terms of extended intersection difference functions (refer section 4.2), which, in turn, are defined in terms of extended intersection functions (refer section 4.1). Intersecting the C curves is relatively uncomplicated because the curves are of the special form: $y = \frac{g(t)}{(1+t^2)^2}$ $(t = \tan(\frac{\theta}{2}))$.

4.1 Extended Intersection Functions $I(\theta, y)$

Extended intersection functions, $I(\theta, y)$ algebraically describe the x coordinate of the extended
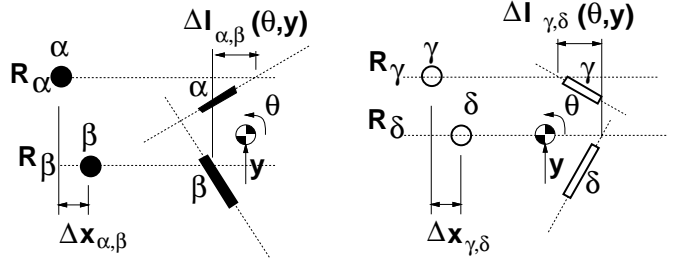

Figure 16: Each relative displacement between two contacts on the same jaw specifies a curve C in (θ, y) configuration space. The contact pose is the intersection of these two curves.

intersection between a horizontal fixture row and a corresponding edge segment $\mathcal{E}_\alpha$ rotated around the reference point by θ and translated by y (refer Figure 17). $I_\alpha(\theta, y)$ is defined in equations (8) and (9) in terms of θ, y, R_α, D_α, α, and t $(t = \tan(\frac{\theta}{2}))$. α is the orientation normal to $\mathcal{E}_\alpha$ pointing inward towards the object's reference point. R_α is the minimum distance from the reference point to $\mathcal{E}_\alpha$, and D_α is the difference in y coordinates between the modular row $\mathcal{R}_\alpha$ and the reference point.
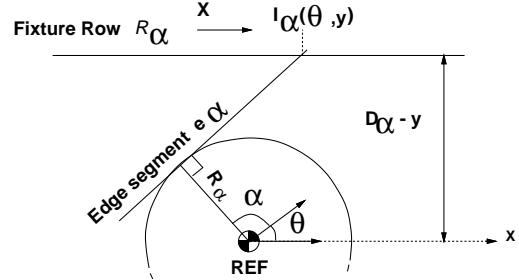

Figure 17: Model used to compute extended intersection $I(\theta, y)$ between edge segment $\mathcal{E}_\alpha$ and fixture row $\mathcal{R}_\alpha$.

$$I_\alpha(\theta, y) = \frac{R_\alpha + (y - D)\sin(\theta + \alpha)}{\cos(\theta + \alpha)} \tag{8}$$

$$= \frac{R_\alpha + (y - D)(\sin\theta\cos\alpha + \cos\theta\sin\alpha)}{\cos\theta\cos\alpha - \sin\theta\sin\alpha}$$

$$I_\alpha(t, y) = \frac{y(-t^2\sin\alpha + 2t\cos\alpha + \sin\alpha)}{(1 - t^2)\cos\alpha - 2t\sin\alpha} + \tag{9}$$

$$\frac{t^2(R_\alpha - D\sin\alpha) - 2tD\cos\alpha + R_\alpha + D\sin\alpha}{(1 - t^2)\cos\alpha - 2t\sin\alpha}$$

4.2 Extended Intersection Difference Functions $\Delta I_{\alpha,\beta}(\theta, y)$

Extended intersection difference functions are defined in equations equations (10) and (11).

$$\Delta I_{\beta,\alpha}(\theta, y) = I_\beta(\theta, y) - I_\alpha(\theta, y) \tag{10}$$

$$\Delta I_{\beta,\alpha}(t,y) = \frac{y((1-t^2)\sin\alpha + 2t\cos\alpha)}{(1-t^2)\cos\alpha - 2t\sin\alpha} +$$

$$\frac{t^2(R_\alpha - D_\alpha\sin\alpha) - 2tD_\alpha\cos\alpha + R_\alpha + D_\alpha\sin\alpha}{(1-t^2)\cos\alpha - 2t\sin\alpha} -$$

$$(\frac{t^2(R_\beta - D_\beta\sin\beta) - 2tD_\beta\cos\beta + R_\beta + D_\beta\sin\beta}{(1-t^2)\cos\beta - 2t\sin\beta} +$$

$$\frac{y((1-t^2)\sin\beta + 2t\cos\beta)}{(1-t^2)\cos\beta - 2t\sin\beta}) \quad (11)$$

4.3 Constant Extended Intersection Difference Curves $C(\theta, y)$

$C_{\beta,\alpha}$ curves include (*necessary, but not sufficient*) configurations (θ, y) satisfying the constraint that the x coordinates of the intersections are separated by exactly $\Delta_{\beta,\alpha}$, and are defined by the zero sets of $C_{\beta,\alpha}(\theta, y)$ functions (refer equations (12), (13)). An algebraic expression is formed by cross multiplying the denominators of $(\Delta I - \Delta)$ $(Q(\theta, y))$.

$$C_{\beta,\alpha}(\theta, y) = (\Delta I_{\beta,\alpha}(\theta, y) - \Delta_{\beta,\alpha})Q(\theta, y) \quad (12)$$

$$C_{\beta,\alpha}(t, y) = y(1+t^2)^2\sin(\alpha - \beta) \quad (13)$$

$$-\Delta_{\beta,\alpha}((1-t^2)\cos\alpha - 2t\sin\alpha)((1-t^2)\cos\beta - 2t\sin\beta) +$$

$$((1-t^2)\cos\beta - 2t\sin\beta)((1+t^2)(R_\alpha - D_\alpha\sin\alpha) - 2tD_\alpha\cos\alpha)$$

$$((1-t^2)\cos\alpha - 2t\sin\alpha)((1+t^2)(R_\beta - D_\beta\sin\beta) - 2tD_\beta\cos\beta)$$

4.4 Intersecting C Curves

This section describes the computation of the orientations t of intersections of constant extended intersection difference curves (without loss of generality, $C_{\beta,\alpha}(t, y) = 0, C\Delta I_{\delta,\gamma}(t, y) = 0$). Observe that the ratio between the y contributions of these curves remains constant over all t. Let C^{*y} refer to the y-independent monomials of C. Cross multiplying the y-dependent and y-independent components of $C_{\beta,\alpha}$ and $C_{\delta,\gamma}$ (refer equation(15)) and dividing out $y(1+t^2)^2$, produces a quartic expression which can be solved numerically.

$$\underbrace{\sin(\alpha - \beta)y(1+t^2)^2 + C^{*y}_{\beta,\alpha}(t)}_{C_{\beta,\alpha}} = 0$$

$$= \underbrace{\sin(\gamma - \delta)y(1+t^2)^2 + C^{*y}_{\delta,\gamma}(t)}_{C_{\delta,\gamma}} \quad (14)$$

$$\sin(\alpha - \beta)C^{*y}_{\delta,\gamma}(t) - \sin(\gamma - \delta)C^{*y}_{\beta,\alpha}(t) = 0 \quad (15)$$

The y translation, as a function of t is computed in three steps: First, rotate the edge segments around the reference point by θ ($\theta = 2\arctan t$), then for each edge segment, compute the line segment of translation vectors which would translate the edge onto the peg position, and finally, intersect the translation lines for all of the same-jaw peg positions. The y component of both jaws' intersections (x, y) should match. σ, the jaw spacing is equal to difference between the x coordinates of the intersections.

5 Conclusion

In this report, we described a complete, efficient algorithm for designing fixture vice configurations for two and a half dimensional objects. The algorithm consists of two routines: enumerating all peg configurations simultaneously contacting four edge segments, and, for each such configuration, computing the contact points and pose of the object. This type of algorithm is a prerequisite for a multi-step fixture design planner, because such a planner must examine all of the valid configurations to find the best one.

References

[1] H. Asada and A. By. Kinematic analysis of workpart fixturing for flexible assembly with automatically reconfigurable fixtures. *IEEE Journal on Robotics and Automation*, RA-1(2):86–94, December 1985.

[2] Randy C. Brost and Ken Y. Goldberg. A complete algorithm for synthesizing modular fixtures for polygonal parts. In *International Conference on Robotics and Automation*. IEEE, May 1994. to be presented.

[3] John Canny and Ken Goldberg. RISC robotics. In *IEEE International Conference on Robotics and Automation*, 1994.

[4] Richard Cole and Chee K. Yap. Shape from probing. *Algorithmica*, 8:19–38, March 1987.

[5] Bernard Faverjon and Jean Ponce. On computing two-finger force-closure grasps of curved 2d objects. In *IEEE International Conference on Robotics and Automation*, number 1, pages 424–429, 1991.

[6] Carlo Ferrari and John Canny. Planning optimal grasps. *IEEE Journal on Robotics and Automation*, pages 2290–2295, 1992.

[7] F. Brack Hazen and Paul Wright. Workholding automation: Innovations in analysis. *Manufacturing Review*, 3(4):224–237, December 1990.

[8] Xanthippi Markenscoff, Lugun Ni, and Christos H. Papadimitriou. The geometry of grasping. *International Journal of Robotics Research*, 9(1):61–74, 1990.

[9] Xanthippi Markenscoff and Christos H. Papadimitriou. Optimum grip of a polygon. *International Journal of Robotics Research*, 8(2):17–29, 1989.

[10] A. Markus, Z. Markusz, J. Farka, and J. Filemon. Fixture design using prolog: An expert system. *Robotics and Computer Integrated Manufacturing*, 1(2):167–172, 1984.

[11] B. Mishra. Workholding: Anaylsis and planning. Technical Report No. 259, New York University, Courant Institute of Mathematical Sciences, November 1986.

[12] B. Mishra, J. T. Schwartz, and M. Sharir. On the existence and synthesis of multifinger positive grips. *Algorithmica*, 2:541–558, 1987.

[13] Van-Duc Nguyen. Constructing force-closure grasps. *International Journal of Robotics Research*, 7(3):3–16, 1988.