Relative Positioning of Planar Parts in Toleranced Assemblies

Yaron Ostrovsky-Berman and Leo Joskowicz
The Hebrew University of Jerusalem, {yaronber,josko}@cs.huji.ac.il

ABSTRACT

Accounting for geometric variability in mechanical assemblies is a key component of modern design methodologies. This paper presents a framework for worst case analysis of the relative position variation of toleranced parts. The framework is based on our general parametric tolerancing model for planar parts whose boundaries consist of line and arc segments and whose vertices are described by standard elementary functions of part dimensions, which vary within tolerance intervals. Here, we present six types of relative position constraints designed to model all types of contact and clearance specifications between features of two parts. For these pairwise constraints, we describe an algorithm to compute the sensitivity matrix of each vertex. This matrix describes the vertex position variation satisfying the constraints as a linear function of the two part parameters. To model the relative part position variation in the entire assembly, we introduce the assembly graph, a generalization of Latombe's relation graph that includes cycles, toleranced parts, and three degrees of freedom. We show how to compute the sensitivity matrices of each vertex from the pairwise relative position constraints and the assembly graph. These matrices serve to compute the tolerance envelopes bounding the areas occupied by the parts under all possible assembly instances. The envelopes are an accurate characterization of geometric uncertainty for assembly planning and mechanism design.

Keywords: variational part models, geometric constraint solving, tolerance envelopes.

1. INTRODUCTION

Manufacturing and assembly processes are inherently imprecise, producing parts that vary in size and form. Tolerance specifications allow designers to control the quality of the production and to manufacture parts interchangeably. Designers prefer tight tolerances to ensure that parts will fit in the assembly and perform their intended function. Manufacturers, on the other hand, prefer loose tolerancing to lower the production cost and decrease the need for quality machine tools and precision measurement machines. Tolerance analysis methods play a key role in bridging between the two.

Predicting the effect of tolerances is difficult even for skilled designers because it requires identifying the critical interactions of toleranced dimensions, which often have complex dependencies. Tolerancing methods have been developed and incorporated into most modern CAD software. However, these methods are limited in the types of interactions they can model and in the quality of the results they produce.

Determining the variations of the relative positioning of parts with tolerances in an assembly is a key problem in assembly planning [3] and mechanism design [9]. For example, nearly all assembly planners produce plans for nominal parts. However, because of shape and position variability due to manufacturing imprecision, the relative part positions vary as well. Thus, the nominal assembly plan might not be feasible for certain instances of parts, and a valid plan for one instance might not be suitable for others. In mechanism design, interference between two part instances can occur even when there is no blocking between the nominal parts.

The relative position of imperfect planar parts was studied by Turner [11], who reduces the problem to solving a non-linear system of constraints for a given cost function. Sodhi and Turner [10] later extended this work for 3D parts. Li and Roy [7] show how to find the relative position of polyhedral parts with mating planes constraints. These methods compute the placement of a single instance of the assembly, and thus cannot be extended to analyze the entire variational class of the assembly. Inui et al. [4] propose a method for bounding the volume of the configuration space representing position uncertainties between two parts. However, their method is only applicable for polygonal parts and is computationally prohibitive. Cazals and Latombe [2] present a simple tolerancing model in which polygonal parts vary in the distance of their edges from the part origin, but not in their orientation. They show how to compute the relative position between two parts when the variational parameters span their allowed range, and use it in assembly planning with infinite translations [6]. They acknowledge the limitations of their model and point to the

need for developing a more general tolerancing model and for supporting other motion types. This motivated our work.

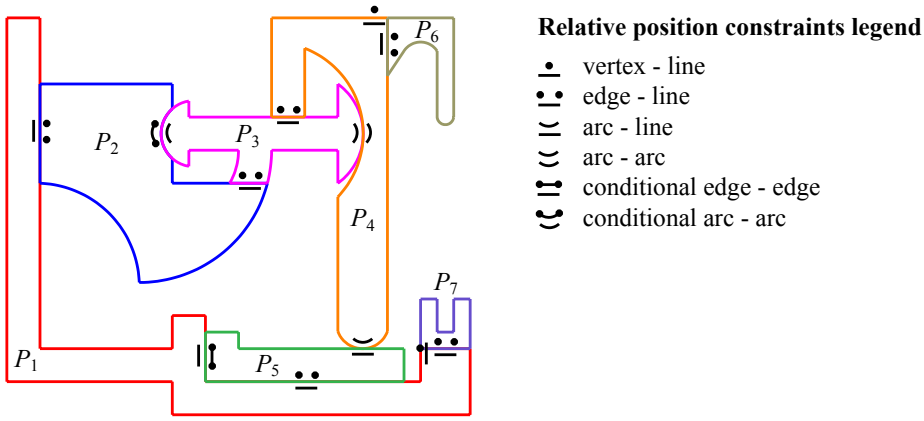


Fig. 1. Example of a simplified seven part planar mechanism with all types of contacts between parts.

In this paper, we present a framework for worst case analysis of the relative position variation of toleranced parts in mechanical assemblies. The framework is based on our previously developed general parametric tolerancing model for planar parts whose boundaries consist of line and arc segments and whose vertices are described by standard elementary functions of part dimensions, which vary within tolerance intervals. We introduce six types of relative position constraints designed to model all types of contact and clearance specifications between features of two parts. For these pairwise constraints, we describe an algorithm to compute the sensitivity matrix of each part vertex. This matrix describes the vertex position variation that satisfies the constraints as a linear function of the two part parameters. To model the relative part position variation in the entire assembly, we introduce the assembly graph, a generalization of Latombe's relation graph that includes cycles, parts with general tolerances, and three degrees of freedom. We then show how to compute the sensitivity matrices of each part vertex from the pairwise relative position constraints and the assembly graph. These matrices are used to compute the tolerance envelopes bounding the areas occupied by the parts under all possible assembly instances. The envelopes provide an accurate characterization of geometric uncertainty that is useful in assembly planning and mechanism design.

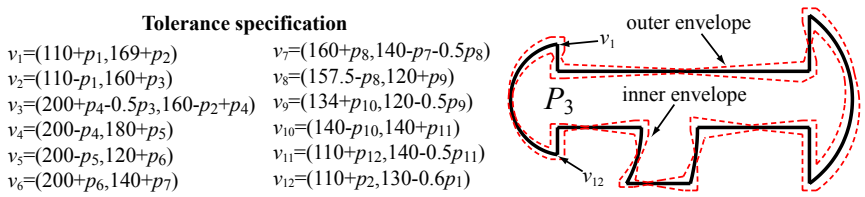
2. TOLERANCED ASSEMBLY SPECIFICATION

Assemblies of toleranced parts require a representation that accounts for part variations. The goal is to develop a framework within which part variations can be represented and efficiently computed. Our starting point is the general model of planar toleranced parts whose boundary consists of line and arc segments that we developed in previous research, which we briefly describe in Section 2.1. In Section 2.2, we describe the six types of relative position constraints between two parts. In Section 2.3, we introduce the assembly graph and describe an algorithm for computing the variability of part positions in the assembly. Throughout the paper, we will use the assembly shown in Figure 1 as an example to illustrate the concepts.

2.1 Toleranced parts

We model part variation with the parametric tolerancing model described in [8]. This model is general, reflects current tolerancing practice, incorporates common tolerancing assumptions, and has good computational properties. In this model, part variation is determined by m parameter values $p=(p_1 \ldots p_m)$, specifying lengths, angles, and radii of part features. The parameters have nominal values and can vary along small tolerance intervals. The coordinates of the part vertices are standard elementary functions of a subset of the m parameters. An instance of the parameter values determines the geometry of the part. Figure 2(a) shows the tolerance specification of part P_3.

In [8], we describe an algorithm for computing the outer and inner tolerance envelopes, which are boundaries of the union and the intersection of all possible parts, respectively. The algorithm inputs the partial derivatives of the vertices according to the m variational parameters, and computes the envelopes under the linear approximation of the model. For a part with n vertices, the algorithm computes the most accurate tolerance envelope in $O(nm^2)$ space and $O(nm^2 \log m)$ time. Figure 2(b) shows the tolerance envelope of part P_3.



(a) (b)

Fig. 2. (a) Tolerance specification and (b) envelope of part P_3. Vertices v_1 to v_{12} are ordered clockwise. Parameters p_1…,p_{12} have all nominal values equal to zero. Typically, the vertex functions are derived from the dimensional tolerance specification, either manually or as output from a symbolic geometric constraint solver. Here we chose the parametrization and tolerance intervals that best illustrate the envelopes properties.

2.2 Relative position of two parts

The relative position of one part with respect to another is modeled with contact and clearance constraints. These constraints specify the location of the part boundaries with respect to each other, or with respect to a reference datum. For planar parts consisting of line and arc segments, the constraints describe how to position a part feature (vertex, edge, or arc) with respect to another one or with respect to a datum (line). The variability of the feature parameters determines the variability of the relative position of parts in the assembly.

It is well known that distance relations between points, lines, and arcs are sufficient to specify all clearance, contact, and angularity relations [1]. We have identified six types of relative position constraints that describe all types of contact and clearance constraints, including simultaneous contacts for deliberate over-constraint. The constraints yield the possible variation in the position of part B (the free part) relative to part A (the fixed part) when B is positioned according to the specification and the variational parameters of both parts span their allowed values. For each vertex u of B, the goal is to compute the transformation matrix that describes the sensitivity of the vertex to variations in the parameters of parts A and B. The $2 \times m$ sensitivity matrix S_u has one column for each of the variational parameters. We first describe the relative position constraints and their associated equations. We then show how to solve the resulting system of equations and compute the sensitivity matrices of B.

2.2.1 Relative position constraints

Planar part B has three degrees of freedom, two for translation and one for rotation. Thus, to uniquely determine its position relative to A, three independent constraints are required. For each instance of the parts, there is a rigid transformation $T = (t_x,t_y,\theta)$ that positions B relative to A and satisfies the constraints. Since the part variations are typically at least two orders of magnitude smaller than the nominal dimensions, the translation offsets and rotation angle are also small. We can thus approximate the transformation with $cos(\theta)\approx 1$ and $sin(\theta)\approx\theta$. For parts whose boundaries consist of line and arc segments, there are six types of distance constraints, with which we can model all contact and clearance specifications: 1. vertex-line; 2. edge-line; 3. arc-line; 4. arc-arc; 5. conditional edge-edge, and; 6. conditional arc-arc. For each constraint type we write in parenthesis the number of degrees of freedom it constrains. The assembly in Figure 1 has all the six types of constraints.

1. *vertex-line constraint* (1): this constraint is used to describe distance and angle relationships between two linear features. For example, in Figure 1, the flush relationship between the top of parts P_6 and P_4 is described with a vertex-line constraint between the left vertex of part P_6 and the line supporting the upper edge of P_4.

   The vertex-line constraint equation is derived as follows. Let $v_i=(v_{ix},v_{iy})$ be a vertex of B and let e_i be an edge of A supported by the line $n_{ix}x+n_{iy}y+c_i=0$, where $n_{ix}^2+n_{iy}^2=1$. Since B undergoes a rigid transformation, the coordinates of v_i are functions of the original coordinates and the transformation variables: $\omega_{ix}=v_{ix}-v_{iy}\theta+t_x$ and $\omega_{iy}=v_{iy}+v_{ix}\theta+t_y$. To constrain the distance between v_i and e_i to be d_i, we write $n_{ix}\omega_{ix}+n_{iy}\omega_{iy}+c_i = d_i$, or:

$$n_{ix}t_x + n_{iy}t_y + (v_{ix}n_{iy} - v_{iy}n_{ix})\theta + v_{ix}n_{ix} + v_{iy}n_{iy} + c_i - d_i = 0 \quad (1)$$

   which is linear in (t_x,t_y,θ). When $e_i\in B$ and $v_i\in A$, the vertices of e_i, denoted e_{i1} and e_{i2}, are transformed by T, and the resulting distance constraint equation is:

$$(e_{i2y}-e_{i1y})t_x + (e_{i1x}-e_{i2x})t_y + ((e_{i1x}-e_{i2x})v_{ix}+(e_{i1y}-e_{i2y})v_{iy})\theta + (e_{i1y}-e_{i2y})v_{ix}+(e_{i2x}-e_{i1x})v_{iy}+e_{i1x}e_{i2y}-e_{i1y}e_{i2x} - d_i = 0 \quad (2)$$

2. *edge-line constraint* (2): this constraint is used to describe a distance relationship between two edges. For example, in Figure 1, the contact relationship between the left edge of P_6 and the right edge of P_4 is described with an edge-line constraint. The edge-line constraints are expressed as two vertex-line constrains, one for each vertex of the edge, with two Equations (1) or (2) (depending on whether the edge belongs to *A* or *B*).

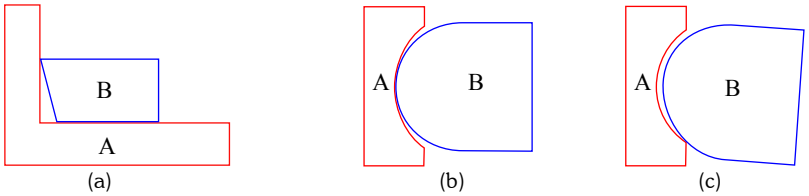


Fig. 3. Conditional constraint cases. (a) contact between the secondary mating edge of *A* and the conditional edge of *B*. (b) contact between the interiors of the conditional arc of *A* and the arc of *B*. (c) contact between the interior of part *B*'s arc and conditional arc endpoint.

3. *arc-line constraint* (1): this constraint is used to describe distance or contact relationships between an arc and an edge, such as the contact between parts P_4 and P_5 in Figure 1. An arc-line constraint entails a linear equation defining the distance of the arc center to the line supporting the edge as the required distance plus the arc radius. In our tolerancing model [8], circular arc segments are specified by the two endpoint vertices v_1, v_2, and either the radius r or the arc angle α, all of which are functions of the variational parameters. The center and radius of the circle supporting the arc is derived from the relations: $c = 0.5(v_1+v_2+v_{12}^{\perp}\tan(0.5(\pi-\alpha))$ and $r = |v_1-v_2|/(2\cos(0.5(\pi-\alpha))$, where $v_{12}^{\perp}$ is counterclockwise perpendicular to v_2-v_1. To derive the constraint equation, we substitute vertex v_i in Equations (1,2) with c_i, the arc center defined by the relation above, and set d_i to be d_i+r. For example, Equation (1) becomes:

$$n_{ix}t_x + n_{iy}t_y + (c_{ix}n_{iy} - c_{iy}n_{ix})\theta + c_{ix}n_{ix} + c_{iy}n_{iy} + c_i - (d_i+r) = 0 \qquad (3)$$

4. *arc-arc constraint* (1): this constraint is used to describe contacts between two arcs, which are common in mechanisms with rotating parts, such as the contact between parts P_3 and P_4 in Figure 1.

The arc-arc constraint entails a quadratic equation setting the distance d between the two arc centers to the required distance plus either $|r_1-r_2|$ (when one arc's supporting circle bounds the other arc, as in parts P_3 and P_4) or r_1+r_2 (when the interiors of the arcs' supporting circles do not intersect). Let c_1 be the center of the fixed part arc and c_2 be the center of the free arc, then the constraint is $(c_{2x} - \theta c_{2y} + t_x - c_{1x})^2 + (c_{2y} + \theta c_{2x} + t_y - c_{1y})^2 = d^2$. We collect the terms according to the transformation variables to get:

$$t_x^2 + t_y^2 + (c_{2y}^2+c_{2x}^2)\theta^2 - 2c_{2y}t_x\theta + 2c_{2x}t_y\theta + 2(c_{2x}-c_{1x})t_x + 2(c_{2y}-c_{1y})t_y + (-2(c_{2x}-c_{1x})c_{2y}+2(c_{2y}-c_{1y})c_{2x})\theta + (c_{2x}-c_{1x})^2 + (c_{2y}-c_{1y})^2 - d^2 = 0 \qquad (4)$$

5. *conditional edge-edge constraint* (1): this constraint is used to specify contacts between nominally parallel edges. In the nominal case, two parallel edges make contact in a line segment; in toleranced assemblies, the contact is usually a point. For example, consider parts P_1 and P_5 in Figure 1. The design intent is to make contact between both pairs of edges: first with the horizontal edges (which are wider and therefore provide more stable contacts), then with the vertical edges. The former edges are termed the primary mating edges, and the latter two are termed the secondary mating edge and the conditional edge (the conditional edge is typically the shorter edge). The secondary contact is generally between the secondary mating edge and a vertex of the conditional edge, but which vertex makes contact depends on the instance of the parts. For example, if the vertical edge of P_1 leans to the right and P_5 is nominal, then the upper vertex of P_5 will be in contact; else, when the edge leans to the left, the lower vertex will be in contact. Both cases yield linear equations between the secondary mating edge's supporting line and one of the two endpoint vertices of the conditional edge as in Equation (1,2). Figure 3(a) shows one of the two conditional cases.

6. *conditional arc-arc constraint* (1): this constraint is used to specify contact between arcs of the same radius. The nominal contact between arcs of the same radius, as in parts P_2 and P_3 in Figure 1, is a circular arc, but when the geometries vary, there are three possible solutions: contact between the interiors of the circular arcs, and contact between an arc and either the first or second endpoint of the other arc, termed the conditional arc. The first case yields a constraint in the form of Equation (4), and the other cases entail a similar quadratic equation with the

conditional arc vertices replacing the arc center. Note that conditional edge-edge and arc-arc constraints specify that contact should occur between the two features, but do not specify which vertex is in contact. We assume that in the nominal case all contacts occur simultaneously, so that any infinitesimal change in the variation parameters either forces the assembly into a state where there is only one contact or leaves the assembly in the same state. This assumption holds in most practical cases.

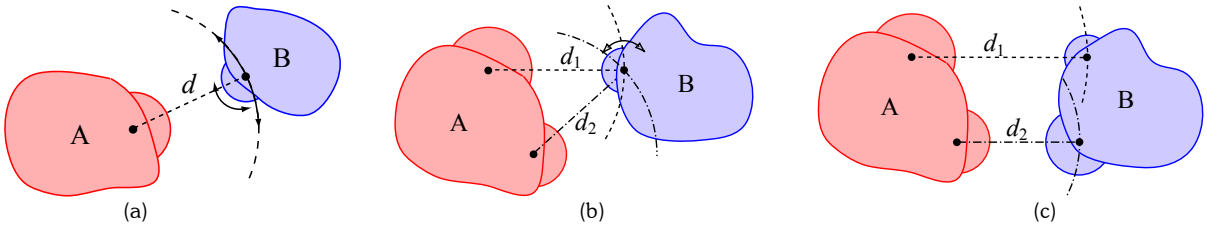


Fig. 4. Modeling arc-arc constraints between the fixed part *A* and the free part *B*. (a) One constraint: part B can rotate about the two arc centers while keeping the arc center at distance *d* from A's arc. (b) Two constraints involving the same arc of B: part B has a rotational degree of freedom. (c) Two constraints involving different arcs of B: the centers of B's arcs are free to move along the dashed circles, while keeping their relative distance fixed.

The six types of relative position constraints can be used to capture the design intent and model all contact and clearance specifications. However, the designer must ensure that the constraints do not yield unsolvable equations. We propose the following modeling guidelines to prevent this situation:

1. Features not participating in the constraints may overlap, even in the nominal solution. The designer should identify them and prevent the overlap by including them in the constraints, as is the case in Figure 1 for the constraint between P_4 and P_5.
2. Lines participating in three constraints must not all be parallel, as this results in dependent equations which are both over and under-constrained. For example, in Figure 1, P_5 cannot simultaneously contact both the left and right edges of the cavity of P_1.
3. Arc-arc constraints may result in equations whose solutions are imaginary. In general, one arc-arc constraint leaves the free part with two rotational degrees of freedom, as illustrated in Figure 4(a). Two rotations about different axes are equivalent to a rotation about the origin, followed by a translation. Since the translation is fixed (it depends on the axes coordinates), the two remaining linear constraints must set the single rotational degree of freedom, so they must be related, as demonstrated by the constraints between parts P_3 and P_4.
   Two arc-arc constraints involving the same arc of part B, as illustrated in Figure 4(b), leave B with a single rotational degree of freedom, which can be set by the third constraint. Two arc-arc constraints involving different arcs of B (Figure 4(c)) allow B's arc centers to move along the circles with distance d_1 and d_2 from A's arcs while keeping their relative distance fixed. The third constraint must determine the position of one arc center on its corresponding circle (dashed), which in turn will fix the position of the second arc center. Three arc-arc constraints are impossible to satisfy with imperfect parts because three circles do not generally intersect in a point.

2.2.2 Computation of the sensitivity matrices

We now present a four-step algorithm to compute the vertex sensitivity matrices. The steps are: 1. model the relations between the two parts; 2. construct the corresponding system of equations; 3. compute the transformation relating the parts and its partial derivatives according to the variational parameters, and; 4. apply the transformation on the vertices of the free part to obtain the sensitivity matrix of each vertex.

In step 1, the relations are determined by three constraints of the six types described in Section 2.2.1. In step 2, the equations are constructed according to Section 2.2.1, and stored in the following abstract form: for linear equations (Eqs (1,2,3)), the form is: $A_i t_x + B_i t_y + C_i\theta + D_i = 0$; for quadratic equations such as Equation (4), the form is: $t_x^2 + t_y^2 + E_i\theta^2 + F_i t_x\theta + G_i t_y\theta + H_i t_x + I_i t_y + J_i\theta + K_i = 0$. For efficiency, we precompute and store the coefficients $A_i,B_i,C_i,D_i,E_i,F_i,G_i,H_i,I_i,J_i,K_i$ and their partial derivatives according to the variational parameters of parts *A* and *B*. There is no need to evaluate the partial derivatives of the coefficient functions, because they can be obtained from the original vertex nominal values and partial derivatives, which were given as input to the algorithm. For example, in Equation (1), $C_i = v_{ix}n_{iy} - v_{iy}n_{ix}$, and its partial derivatives are $\partial C_i/\partial p_j = \partial v_{ix}/\partial p_j * n_{iy} + \partial n_{iy}/\partial p_j * v_{ix} - \partial v_{iy}/\partial p_j * n_{ix} - \partial n_{ix}/\partial p_j * v_{iy}$. Since the normal (n_{ix},n_{iy}) is also a function of the original vertices, its partial derivatives can also be precomputed.

For the equations of the circular arc constraints, the partial derivatives of the coefficients with respect to the variational parameters depend on the parametrization of the arc. When the arc is defined by the angle α, the radius changes according to the above relation, i.e., the constraint distance has non-zero derivatives. When the arc is defined by the radius r, the angle depends on the variational parameters, i.e., the position variation of the arc center also depends on the partial derivatives of α.

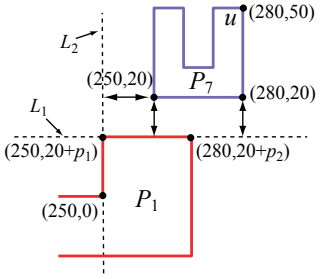


Fig. 5. Relative position constraints between parts P_1 and P_7. Distance constraints are denoted by two headed arrows (distances are zero). The horizontal edges are forced to overlap by an edge-line contact constraint, and their left endpoints are forced to coincide by a vertex-line contact constraint.

In step 3, we solve the system of abstract equations constructed in step 2 by substituting the coefficients into general solution templates that we derived for the following three types of systems: 1. three linear equations; 2. two linear equations plus one quadratic equation; 3. two quadratic equations plus one linear equation.

The general solution to the system of three linear equations is:

$$\begin{aligned} t_x &= (B_1C_3D_2\text{-}B_1C_2D_3\text{+}C_1B_2D_3\text{-}C_1D_2B_3\text{+}D_1C_2B_3\text{-}D_1C_3B_2) \,/\, (A_2C_1B_3\text{-}A_1C_2B_3\text{-}C_3A_2B_1\text{-}A_3C_1B_2\text{+}C_2A_3B_1\text{+}A_1C_3B_2) \\ t_y &= (C_3A_2D_1\text{-}A_2C_1D_3\text{-}C_3A_1D_2\text{+}A_1C_2D_3\text{+}C_1A_3D_2\text{-}C_2A_3D_1) \,/\, (A_2C_1B_3\text{-}A_1C_2B_3\text{-}C_3A_2B_1\text{-}A_3C_1B_2\text{+}C_2A_3B_1\text{+}A_1C_3B_2) \\ \theta &= (A_3B_1D_2\text{-}A_3D_1B_2\text{+}B_3A_2D_1\text{-}B_3A_1D_2\text{+}D_3A_1B_2\text{-}D_3A_2B_1) \,/\, (A_2C_1B_3\text{-}A_1C_2B_3\text{-}C_3A_2B_1\text{-}A_3C_1B_2\text{+}C_2A_3B_1\text{+}A_1C_3B_2) \end{aligned} \quad (5)$$

The system of equations resulting from $0 \le z \le 2$ arc-arc (quadratic) constraints and 3-z linear constraints has 2^z solutions. However, only one of them corresponds to the nominal positions of the parts. The correct solution is identified by comparing the transformed vertices with the nominal vertex positions. The general solutions for systems with quadratic equations were derived using MAPLE, and are too lengthy to reproduce here.

For the partial derivatives of each of the template solutions, we derived corresponding templates, consisting of the coefficients and their partial derivatives. Since these were computed in step 2, the nominal solution $T = (t_x,t_y,\theta)$ and its derivatives $\partial T/\partial p_j = (\partial t_x/\partial p_j, \partial t_y/\partial p_j, \partial\theta/\partial p_j)$ are computed with a constant number of elementary arithmetic operations.

In step 4, we use the transformation derivatives to compute the sensitivity matrices of the vertices of B. Each vertex $u \in B$ undergoes the transformation T in order to satisfy the relations. Thus, its nominal position is: $\omega_x = u_x - u_y\theta + t_x$ and $\omega_y = u_y + u_x\theta + t_y$. Its partial derivatives according to the variational parameters of both A and B define the sensitivity matrix S_u, whose j^{th} column is:

$$\frac{\partial \omega}{\partial p_j} = \left(\frac{\partial u_x}{\partial p_j} - \frac{\partial u_y}{\partial p_j}\theta - \frac{\partial \theta}{\partial p_j}u_y + \frac{\partial t_x}{\partial p_j}, \frac{\partial u_y}{\partial p_j} + \frac{\partial u_x}{\partial p_j}\theta + \frac{\partial \theta}{\partial p_j}u_x + \frac{\partial t_y}{\partial p_j} \right)^T \quad (6)$$

Note that the transformation of the vertices is sufficient to describe the segments of the transformed part, as line segment are completely defined by their endpoints, and the radii and angles of circular arcs remain constant under rigid body transformations.

When one of the constraints is conditional, the transformation T, which is correct for all instances of the assembly, is computed as follows. First, we solve the system of equations once for each of the cases (two or three solutions), and denote the resulting transformations $T_i = (t_{ix},t_{iy},\theta_i)$, $1 \le i \le 3$. An infinitesimal change in a single parameter p_j either results in one of T_i being the correct solution, or leaves all solutions correct. In the latter case, $\partial T/\partial p_j = \partial T_1/\partial p_j = \partial T_2/\partial p_j = \partial T_3/\partial p_j$. In the former case, we first determine which of the solutions is correct (by checking distance relations) for an infinitesimal increase and decrease of p_j, and denote it by T_j^+ and T_j^-, respectively. We then compute the left-hand and right-hand derivatives as follows: $\partial T^+/\partial p_j = \partial T_j^+/\partial p_j$, and $\partial T^-/\partial p_j = \partial T_j^-/\partial p_j$, where $\partial T_j^+/\partial p_j$ ($\partial T_j^-/\partial p_j$) is the right-hand (left-hand) derivative of T_j. The algorithm for computing tolerance envelopes in [8] can be used directly

with a straightforward modification to handle input divided into left and right-hand derivatives, at no asymptotic extra cost.

Note that the combined transformation T is accurate when a single parameter varies, but may not be accurate when several parameters vary simultaneously. This is a limitation of the linear approximation, which sums the effects of all the parameters and ignores dependencies. However, the corresponding tolerance envelope is conservative, never underestimating the worst case variation.

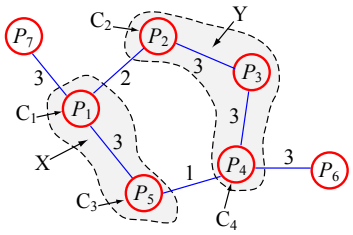


Fig. 6. The assembly graph of the example in Fig. 1. The distances between part features that are connected by an edge are zero. Parts C_1, C_3 and C_2, C_4 of rigid bodies X and Y, respectively, contain features constraining the relative position of the two bodies.

2.2.3 Example

We illustrate the relative position computation on parts P_1 and P_7 in Figure 1 where P_1 is toleranced as shown in Figure 5 (for simplicity, P_7 is not toleranced).

In step 1, we specify an edge-line constraint and a vertex-line constraint designed to get a flush relationship between the parts (distances are zero). In step 2, we construct the equations based on the lines supporting the edges. The equations of lines L_1 and L_2 are, respectively: $(p_1\text{-}p_2)x+30y+250p_2\text{-}280p_1\text{-}600 = 0$ and $(-20\text{-}p_1)x+250p_1+5000 = 0$ (the line equations are not normalized, but since the constraint distances are zero this does not matter). Substituting the coefficients into Equation (1) we get the following system of linear equations:

$(p_1 - p_2)t_x + 30t_y + (7500 - 20p_1 + 20p_{2)}\theta - 30p_1 = 0$
$(p_1 - p_2)t_x + 30t_y + (8400 - 20p_1 + 20p_2)\theta = 0$
$-(p_1 + 20)t_x + (400 + 20p_1)\theta = 0$

In step 3 we substitute the coefficients into Equation (5) to get the nominal solution $\{t_x = 2/3(p_2 - p_1),\ t_y = (28/3p_1 - 25/3p_2,\ \theta = 1/30(p_2 - p_1)\}$. Notice that when $p_1=p_2$, no rotation or horizontal translation is required. We substitute the coefficients and their derivatives into the derivative template and get:
$\partial T/\partial p_1 = (-2/3,\ 28/3,\ -1/30)$ and $\partial T/\partial p_2 = (2/3,\ -25/3,\ 1/30)$. We illustrate step 4 on the vertex $u=(280,50)$. Substituting the computed transformation derivatives and the vertex coordinates into Equation (6), we get the sensitivity matrix $S_u = \begin{pmatrix} 1 & -1 \\ 0 & 1 \end{pmatrix}$.

2.3 Relative positions of parts in an assembly

We now describe how to model the relative position of parts in the entire assembly. Previous work by Latombe et al. [6] introduces the *relation graph* to describe the relative position constraints between nominal parts with two degrees of freedom each. We extend this graph to include cycles and support parts with general tolerances and three degrees of freedom, and call it the extended relation graph, or simply the *assembly graph*.

Graph nodes correspond to parts and undirected edges correspond to constraints between parts. Edge weights are 1, 2, or 3, and indicate the number of degrees of freedom constrained for relative positioning of the two parts. The edge data structure holds additional information about each constraint, such as the feature names of parts A and B, the value or parametric expression of the distance between these features, and the type of constraint. Figure 6 shows the assembly graph of the assembly in Figure 1.

The assembly specification is properly-constrained if it is both complete and non-redundant. The specification is complete if the relative position of all pairs can be determined from the constraints. It is non-redundant if the removal of any constraint results in incompleteness. A necessary and sufficient condition for a properly-constrained assembly of N parts is that the sum of edge weights is $3(N - 1)$, and that for each cycle in the graph with N_c nodes, the sum of weights is $3(N_c - 1)$ and there is exactly one edge of weight 2 and one edge of weight 1 (a cycle with three edges of

weight 2 results in a non-linear system of six equations with no general solution). The above conditions are a special case of the Grübler equation for planar mechanisms [12]. Properly-constrained assembly graphs have two important properties:

1. When two parts are connected by a chain of edges of weight 3, their relative position is determined link by link, where each link is solved as in Section 2.2. Such a chain of parts can be regarded as a single rigid part, because any rigid transformation on the parts as a group preserves the relation constraints.
2. A cycle of N_c parts has exactly N_c-2 edges of weight 3, one edge of weight 2, and one edge of weight 1. The last two edges divide the cycle into two non-intersecting sets of parts X and Y connected by the edges. The relative position between parts connected by edges of weight 1 or 2 cannot be determined because it is under-constrained, but the relative positions between the rigid bodies corresponding to X and Y is well constrained. Figure 6 shows the sets X, Y for the assembly in Figure 1.

Input: Assembly graph, toleranced parts models
1. Find a path in the assembly graph between P_i and P_j.
2. Iterate on the path edges $e = (P_k, P_l)$ in order:
   If weight(e) = 3 **then** compute transformation T_{kl} positioning P_l relative to P_k (Section 2.2).
   Else if weight(e) < 3 (cycle edge) **then**
   i. Find rigid bodies X and Y from graph cycle (X contains P_k).
   ii. Identify parts with constrained features C_1,C_2,C_3,C_4 (as in Figure 6).
   iii. Compute transformations positioning parts in X relative to P_i.
   iv. Compute transformations positioning parts in Y relative to C_2.
   v. Compute transformation T_{XY} positioning Y relative to X according to constraints in C_1,C_2,C_3,C_4.
   vi. Continue path from the exit edge (if it exists)
3. **For** each $u \in P_j$
   a. **For** each variational parameter p_k in parts from P_i to P_j
      i. Find the two transformations that depend on p_k.
      ii. Apply each transformation on u using Equation (6).
      iii. Sum the derivatives of the previous step for the k^{th} column of S_u.

Output: Sensitivity matrices of part vertices

Table 1 shows the algorithm to compute sensitivity matrices of part P_j relative to P_i . Its inputs are the assembly graph and the toleranced part models, including initial vertex partial derivatives. The sensitivity matrices of the vertices of P_j depend on the variational parameters of all the parts in the path from P_i to P_j. The algorithm computes the relative position transformations between pairs of parts on the path from P_i to P_j. The transformation T_{XY} positioning the two sets of cycle parts is applicable for each part in Y. Since the parameters of a part participate in two transformations at most, there are at most two transformations which have non-zero k^{th} columns in the sensitivity matrices of the vertices of P_j. The algorithm applies only the appropriate transformations when computing the matrix columns.

The complexity of the algorithm is as follows. Let r_{ij} denote the number of parts in the path from P_i to P_j (including cycle parts). Let q_k be the maximal number of variation parameters affecting a vertex of the k^{th} part, and let $q = \max_k\{q_k\}$. At each iteration of step 2, we compute the transformation $T_{kl} = (t_x,t_y,\theta)$ between two parts and its partial derivatives according to the local variation parameters. There are up to 12 vertices participating in each set of constraints (when there are three arc-edge or arc-arc constraints), so the number of local parameters is at most $12q$. From Section 2.2, the solution for T_{kl} has constant size, and so do each of its $O(q)$ derivatives. The n_j vertices of P_j depend on at most $12qr_{ij}$ variation parameters, so the computation of their sensitivity matrices in step 3 takes $O(n_jqr_{ij})$ time.

Note that this result is a generalization of the result of Cazals and Latombe [2] for parts with two degrees of freedom. In their tolerancing model, only translations are used to satisfy the constraints. This means that all the vertices of P_j are translated uniformly, and the translation depends on $O(qr_{ij})$ parameters. The tolerance zone of the translation is a convex polygon inside which the origin of part P_j varies. It can be computed in $O(qr_{ij}\log(qr_{ij}))$ time (see [8]), and since q is constant (at most six), this compares with their result. Note also that in their model, the vertices are linear functions of the variation parameters, so the approximation is in fact exact.

One limitation of our method is that while the linear approximation is good for small variations, long chains of mated parts may cause non-negligible cumulative errors. However, in the assemblies that we studied, with chains of up to 8 parts, we did not observe this problem.

3. APPLICATIONS AND EXAMPLE

The sensitivity matrix of a vertex in a toleranced assembly describes the effect of the parameter variations on the position of the vertex, relative to the chosen source part, or the datum. When used as input to the algorithm in [8], the resulting tolerance envelope bounds the convex polygon inside which the vertex is located under all possible variations of the assembly parts. Similarly, the tolerance envelope of the entire part bounds the area occupied by the part under all possible assembly variations. In functionality analysis of mechanisms, this provides a worst case bound on variability of clearances and critical features.

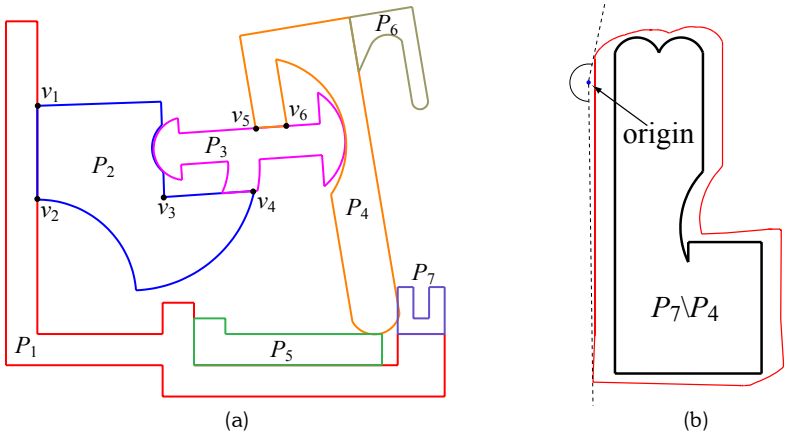


Fig. 7. (a) Infeasible instance of the mechanism when the coordinates of vertices v_1,v_2,v_3,v_4 of P_3 and v_5,v_6 of P_4 vary in 1*mm* from the nominal. To satisfy the constraints, the parts forming the cycle make much larger variations in their positions, causing P_4 and P_7 to collide. (b) The configuration space obstacle $P_7 \setminus P_4$ (thick curve), its tolerance envelope (thin curve), and the cone of blocked translation directions.

One very useful property of the sensitivity matrices is their additivity – it is possible to combine matrices of vertices with shared parameter dependency to obtain the correct combined sensitivity. Without respecting parameter dependencies, the stack-up analysis of feature tolerance zones is overly conservative, as it ignores parameters whose effect on the variability of two features cancel each other out. The additivity is especially important in the computation of configuration spaces (C-space) of toleranced assemblies [9].

In the C-space approach for assembly analysis [3,4,5,9], the space describing the degrees of freedom of a part or a group of parts is partitioned into free space, corresponding to valid configurations, and blocked space corresponding to, invalid configurations caused by overlap with other parts in the assembly, which are treated as obstacles. For motion planning with limited translations, the C-space of a part P_i is two dimensional, and the obstacle made by part P_j is computed using the Minkowski difference of sets: $P_j \setminus P_i = \{v_j - v_i \mid v_i \in P_i,\ v_j \in P_j\}$. The outer boundary of the obstacle is obtained by first computing the boundary features (vertices and line and arc segments in our model) pairwise Minkowski difference and then computing the outer cell of the resulting arrangement of curves. This outer bounds the C-space obstacle of the parts.

When the parts are toleranced, we can first compute each part's tolerance envelope, and then compute the configuration space obstacle using the envelopes. However, this analysis is overly conservative because it ignores parameter dependencies. The correct method is to construct the *C-space envelopes* for pairs of parts as follows. First, compute the pairwise feature Minkowski difference as before. The segments bounding the Minkowski difference are either the difference of a vertex of P_i and a segment of P_j (or vice versa), or parts of such segments produced by the intersection of two segments. In either case, the vertices of these segments have explicit representations as functions of the vertices of P_i and P_j. Thus it is possible to compute their nominal positions, and their partial derivatives, which are a linear combination of the sensitivity matrices of P_i and P_j. Next, we compute the outer tolerance envelopes of the pairwise features, and obtain an arrangement of elementary features, whose outer cell is the C-space envelope of the

obstacle. When C-space envelopes of obstacles are used in assembly analysis methods instead of the nominal obstacles, the analysis accounts for parameter dependency in the worst case variation of the parts in the assembly.

We have implemented the relative position computation and assembly graph data structure using MATLAB, and ran it on several examples of assemblies. To illustrate, we describe the results we obtained on the assembly in Figure 1. The input consists of $N=7$ parts, $m=58$ variational parameters. The maximum part complexity is $n=14$, maximal path length $r=7$ (from P_6 to P_7 and vice versa, including cycle parts), and maximal number of local parameters $q=3$. The CPU time to compute the sensitivity matrix of all the parts relative to P_1 was 1.37 seconds on a Pentium IV 2.4GHz with 512 MB RAM. Figure 7(a) shows an instance of the mechanism when only six of the variational parameters are allowed to vary within $\pm 1mm$ tolerance intervals (about 1% from the average feature length in the assembly). Even though the nominal horizontal clearance between P_4 and P_7 is of $20mm$ and four of the vertices translate vertically, the instance represents an infeasible assembly because the parts overlap.

Even with smaller tolerance intervals, tolerancing significantly affects the assembly. Figure 7(b) shows part P_7 as the configurations space obstacle of P_4, without tolerances, and when each variational parameter has a $\pm 0.3mm$ tolerance interval. The obstacle and the origin determine the directions in which P_4 is free to move without colliding with P_7 (shown in Figure 7(b) as part of the unit circle). Thus, unlike the nominal assembly, there are instances of the assembly in which of P_7 blocks the directions separating P_4 from P_3, and therefore P_7 must be removed first in the disassembly sequence.

4. CONCLUSION

We have presented a framework for worst case analysis of toleranced planar assemblies. The framework is more general than existing ones in terms of the geometry of parts (line and arc segment boundaries) and the tolerancing model used. The sensitivity matrices of each part vertex that are computed by our algorithm can used to compute the tolerance envelopes bounding the areas occupied by the parts under all possible assembly instances. The envelopes provide a characterization of geometric uncertainty that is more accurate than those produced by Monte Carlo methods and is useful in assembly planning and mechanism design. Directions for future work include modeling mechanisms of interest to industry, extension to three-dimensional parts, starting with polyhedra, and optimal part placement with respect to objective functions [7, 10] instead of pre-determined contacts.

5. REFERENCES

[1] Bouma, W., Fudos, I., Hoffmann, C.M., Cai, J., and Paige, R. Geometric constraint solver, *Computer Aided Design*, Vol. 27(6), pp 487-501, 1995.

[2] Cazals, F. and Latombe, J.-C., Effect of tolerancing on the relative positions of parts in an assembly, *IEEE International Conference on Robotics and Automation*, 1997.

[3] Halperin, D., Latombe, J.-C., and Wilson, R.H., A general framework for assembly planning: The motion space approach, *Algorithmica*, Vol. 26, pp 577-601, 2000.

[4] Inui, M., Miura, M. and Kimura, F., Positioning conditions of parts with tolerances in an assembly, *IEEE International Conference on Robotics and Automation*, 1996.

[5] Latombe, J.-C., *Robot motion planning*, Kluwer Academic Publishers, 1991.

[6] Latombe, J.-C., Wilson, R.H., Cazals, F., Assembly sequencing with toleranced parts, *Computer-aided Design*, Vol. 29(2), pp 159-174, 1997.

[7] Li, B. and Roy, U., Relative positioning of toleranced polyhedral parts in an assembly, *IIE Transactions*, Vol. 33(4), pp 323-336, 2001.

[8] Ostrovsky-Berman, Y. and Joskowicz, L., Tolerance envelopes of planar mechanical parts with parametric tolerances, *Computer Aided Design,* Vol. 37(5), pp 531-544, 2005.

[9] Sacks, E. and Joskowicz L., Parametric kinematic tolerance analysis of general planar systems, *Computer Aided Design*, Vol. 30(9), pp 707-714, 1998.

[10] Sodhi, R. and Turner, J.U., Relative positioning of variational part models for design analysis, *Computer Aided Design*, Vol. 26(5), pp 366-378, 1994.

[11] Turner, J.U., Relative positioning of parts in assemblies using mathematical programming, *Computer Aided Design*, Vol. 22(7), pp 394-400, 1990.

[12] Whitney, D.E., *Mechanical Assemblies: Their Design, Manufacture, and Role in Product Development*, Oxford university press, 2004.