Proceedings of DETC'01
ASME 2001 Design Engineering Technical Conferences and Computers and Information in Engineering Conference
Pittsburgh, Pennsylvania, September 9-12, 2001

DETC2001/DAC-21107

CAPTURING ASSEMBLY TOLERANCES AND CRITERIA IN A COMMON MODEL

Edward P. Morse
Center for Precision Metrology
Department of Mechanical Engineering and Engineering Science
University of North Carolina at Charlotte
Charlotte, NC 28223-0001

ABSTRACT

Assemblies of mechanical parts are typically modeled as collections of rigid solids, constrained by contact conditions at "mating" faces. We have previously proposed a method – called *GapSpace* – of modeling and representing assemblies with constraints arising from either face adjacency or face contact. This method of specification allows less restrictive tolerances to be placed on the assembly components, resulting in specifications that are less expensive to meet. In this paper we show how various assembly metrics and criteria can be represented with GapSpace, and how to analyze toleranced assemblies for conformance to these criteria. These analyses show quantitatively the importance of deciding *which* part dimensions will have tolerances applied to them, in addition to the choice of the actual tolerance values.

1 INTRODUCTION

This paper addresses assemblies of rigid parts where relative motion occurs in a single direction, and the physical interaction between parts occurs at faces perpendicular to the motion direction. We refer to the analysis of such assemblies as one-dimensional. This type of analysis is useful because many assemblies and sub-assemblies, including those with rotational symmetries, may be abstracted to one-dimension for analysis. Two such assemblies are shown in Figure 1 below. The first example, the disk brake and caliper assembly, consists of five parts and has a single clearance (or "gap") condition. The second example, the hinge, has only two parts but four clearance conditions.

The specification of assemblies in this paper relies on the ability to describe feature adjacency either *with or without* a contact requirement. Here we use the term feature to describe a subset of the part's surface. The disk brake assembly in Figure 1 has a series of parts in contact with one gap that depends on the component dimensions, while the hinge has four feature *adjacencies* required for successful assembly, but contact between a specific pair of features is not required. This second method of specification differs from that used by most CAD packages, where explicit placement of the individual parts is required. The "floating assemblies" [1] which arise from specifications consisting only of adjacencies cannot be represented by the part placement assembly techniques these packages employ.

In this paper we present a brief overview of the GapSpace modeling method introduced in [2] and [3], and then describe various assembly criteria in the GapSpace framework. Finally, the analysis of toleranced assemblies for conformance to these criteria is examined.

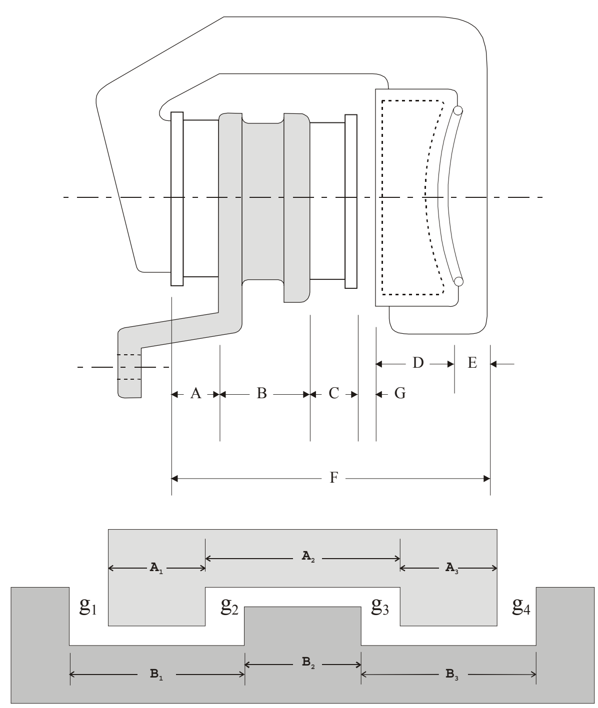


Figure 1: Assemblies amenable to 1-dimensional analysis.

2 RELATED WORK

The representation of variability in mechanical parts and assemblies may be divided into the following classes: minimal parametric, geometric, and extended parametric [4]. This list is not necessarily complete; other representations may emerge. In all but a few cases, the one-dimensional analysis of assemblies is contained in the first (minimal parametric) class, although it is possible to define one-dimensional geometric tolerances that cannot be described using a minimal parametric representation.

2.1 Part variability

The composition of worst-case parametric tolerances in one dimension through linear accumulation, or "stack-up," is a fundamental technique in mechanical engineering. While the method seems so obvious as to be stated simply as "fact" in many works, there are a few researchers who have described stack-up well. Requicha [5] observes that dimensioning and tolerancing practice requires that we apply dimensions to individual parts so that a "dimension tree" is formed. We follow this practice with only minor modification (the use of directed arcs in the trees, instead of explicitly choosing a root node). Although many who reference Requicha's work to justify the use of dimension trees do not mention it, we note here Requicha's (paraphrased) statement about dimension trees: while the tree structure follows industrial practice, it is not clear that an "over-dimensioned" graph is bad, and it may be a useful way to specify parts, especially if closed-loop control is available in the manufacturing process (i.e. we can measure the part "in process" and base the next machining operation on the measurement result.) Turner and Wozny [6] also point out that over-dimensioning can add "meaningful new constraints" to the tolerance specification. One recent work by Dorminey [7], based on the GapSpace concepts described in this paper, explores the specific advantages of over-dimensioning.

2.2 Assembly variability

The stack-up of tolerances on parts is extended to assemblies by requiring that the assembly components be joined at certain places. Early work in this area by Bjørke [8] (originally published in 1978) describes the closure of tolerance chains with a sum dimension. These sum dimensions are typically clearances or gaps which must accommodate the variability accumulated in the tolerance chain. Bjørke also uses a tree structure for parts and assemblies, and relies on the sum dimensions between features to create cycles, resulting in assembly graphs. Another description of linear stack-up is found in Juster [9], where a graph model is used to automate worst-case analysis across assemblies.

2.3 Higher-dimensional extensions

The primary challenges in extending linear analysis to 2- and 3-dimensional problems are the introduction of imperfectly formed features, and the non-linear relationships which result from the possibility of rotations and the interaction of curved features. Two primary techniques are used to manage this complexity. The first assumes that all features have perfect form, and tracks the variability of part contacts based on size changes of the parts and the kinematics of part pairings. The second technique is to use additional parameters to describe features (e.g. we might model a planar surface with a polynomial of degree ≥ 2.)

Bjørke uses the first technique, as does a group of researchers at Brigham Young University working under Chase; the latter have produced a large body of work based on the "vector loop" approach to tolerance analysis [10][11][12]. This method – a direct extension of linear stackup – identifies chains of dimensions (represented as vectors) on 2- and 3-D parts which contribute to critical gap dimensions. The equations relating these loops are solved in each "dimension" and a resulting set of gap values produced.

A kinematic technique that is becoming increasingly well-known – called TTRS – is based on the work of Clement [13]. This method classifies all inter-feature relationships into classes corresponding to low-order kinematic pairs. While surface contact is not required for many of these relationships, the relationships imposed in the classification (coincidence or parallelism of cylinder axes, for example) impose constraints on assembly analysis that are similar to the constraints imposed by surface contact.

The second technique, higher-order parameterization of contacting surfaces, appears in one commercial software package as "Monte Carlo analysis." In this implementation, points are selected on the contact surfaces and allowed to vary randomly to imitate form errors on candidate parts. For each set of random variations, an analysis is performed to determine the resulting influence on some clearance or gap dimension. After many such analyses, an estimate of a distribution representing the clearance dimension is made.

2.4 Exceptions

In all of the methods described above, the tolerances are accumulated across *contacts* between assembly components, and a single resultant gap analyzed for each cycle on a graph representation of the assembly. This is distinctly different from the GapSpace model, where assemblies may or may not have contacts explicitly enforced, and cycles of toleranced dimensions may contain more than one gap.

Notable exceptions to the "single gap" approach described above appear in Fleming's thesis [14] and the work of Mullins and Anderson [15]. Fleming assumes that parts in assemblies are able to move freely, and are prohibited from having other than "potential contact" with other parts. Mullins and Anderson have developed the concept of a physically constraining face set (PCFS), which bounds the motion of a part with respect to the other parts in the assembly. Parratt's [1] introduction of the *generalized feature of size*, made up of any pair of opposing features on a part, and the observation that traditional stackup analysis may be overly conservative led us to examine the dependencies between the necessary and sufficient conditions for assembly more closely. This research ultimately led to the development of GapSpace.

3 GAPSPACE OVERVIEW

In this section we provide a short explanation of GapSpace, and how assemblies may be modeled and analyzed in this space. At its current stage of development, GapSpace is used to analyze assemblies in one direction at a time. The abstraction of solids for one-dimensional analysis and the fundamentals of GapSpace are described in [2], and we review the material briefly here.

3.1 Modeling and Representation

We represent rigid mechanical parts with r-sets. The term r-set, as introduced in [16], describes a bounded, closed, regular, semi-analytic set. For one-dimensional assembly analysis, parts are modeled as *layered* one-dimensional r-sets. The layering extends the domain of one-dimensional r-sets so that more complex parts and assemblies may be modeled. In Figure 2 we show a model of a three-dimensional part that has three faces (F_1, F_2, and F_3) that may come in contact with other parts in the assembly. In a strictly one-dimensional model, the faces would be modeled as points on a line, and the interior of the part would be modeled with an interval connecting the points, as shown in Figure 2a, and information about F_2 would be lost. The layered model resolves this problem. Figure 2b shows a layered model, based on the *evaluation rays* shown in Figure 2a. While we don't state an explicit algorithm for choosing evaluation rays (which are parallel to the modeling space), each face must be pierced by at least one evaluation ray. The result of the layering is that Face 1 appears on both layers (as point P1), while Faces 2 and 3 each appear on a single layer.

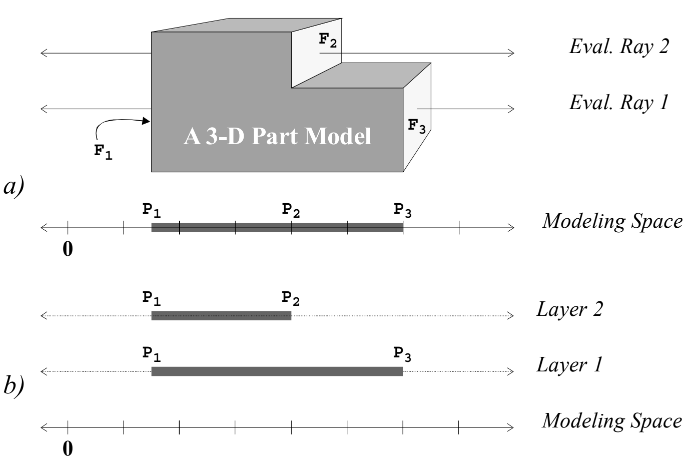


Figure 2: The layered 1-D model for 3-D parts.

The representation of the models described above is accomplished using a one-dimensional boundary representation which has a tree structure. As mentioned in Section 2.1, this use of trees to represent parts is nearly universal. Each feature relevant to assembly is represented as a node, with the distance between feature pairs represented with an arc. We use directed arcs to represent dimensions, and assign a signed distance value to each arc.

A tree representation of the part in Figure 2 is shown in Figure 3. In our trees, the directed arcs indicate the ordering used for the signed distance value of the arc. Each feature node also contains information identifying the "non-material" side of the feature (Face 1 is a *left feature*, while Faces 2 and 3 are *right features*).

We can compute the signed distance between any ordered pair of features $\{F_1, F_2\}$ in the tree by traversing the tree from F_1 to F_2 and accumulating the signed distances along each arc of the tree. Accumulation of signed distances follows this rule: if the traversal is in the direction of an arc, the signed distance value is added and if not, subtracted. Note that the direction of the arcs does not control the traversal direction; we may traverse either of the graphs in Figure 3 from F_1 to F_2.

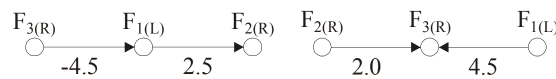


Figure 3: Two tree representations of a part using signed distances and directed arcs.

Assemblies are modeled as collections of parts with additional information about how these parts are to interact. In order to perform assembly analysis, our model must be able to capture the following relationships between parts: contact, interference, and absolute or relative position.

Assembly representation is accomplished by creating a graph from a forest of part trees. This graph is formed by adding directed arcs that correspond to pairings of surfaces on different parts, which we call *liaisons*. These liaisons are identified by the designer as features that are adjacent in the final assembly, and are drawn in the increasing (positive) direction of the modeling space so that a positive liaison distance corresponds to space (a gap) between the parts.

The features comprising a liaison may be required to be in contact, in which case the liaison describes a *mating condition* (MC.) If contact is not required, the liaison describes a *clearance condition* (CC.) Informally, we refer to the mating condition liaison as a "contact" and the clearance condition liaison as a "gap." We say a liaison condition is honored when the liaison distance is equal to zero for MCs, or when the distance is greater than or equal to zero (non-negative) for CCs.

As the designer creates an assembly specification, the necessary liaisons will probably appear as single relations ("feature a_3 of part A *goes against* feature b_2 of part B") or pairs of relations ("feature pair a_5, a_6 *fits in* feature pair b_8, b_9"). Useful requirements for assemblies are that the parts don't interfere, and that the relative motion between parts is adequately controlled. In this paper we assume that the complete specification of liaisons is provided by the designer as part of the assembly definition. We could, perhaps, infer liaisons from the proximity of part features in a nominal assembly representation, but this procedure would be difficult to automate.

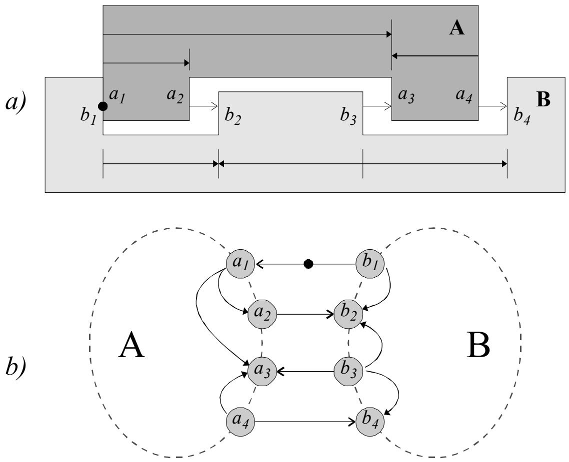


Figure 4: An assembly and its part trees joined as a graph.

After liaisons have been specified by the designer, the dimension trees are joined into a graph. As seen in Figure 3, the dimension arcs within a tree are tied to the choice of sign for the "directed dimensions" and can thus point in either direction. The liaison arcs, however, must be directed from a right feature to a left feature. This is to maintain consistency between the liaison conditions so a positive liaison distance will always correspond to a gap between the parts. Figure 4a shows an assembly of two parts related by one mating condition and three clearance conditions. In Figure 4b the two part trees are enclosed in dashed rings to differentiate between the parts, while the liaison arcs connect the parts.

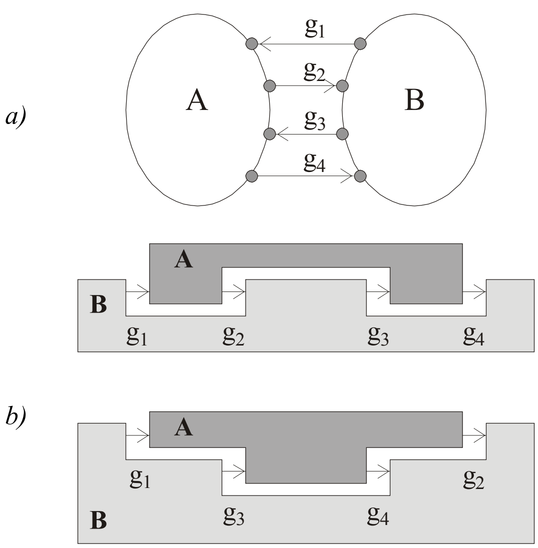


Figure 5:A liaison graph and two of the assemblies it represents.

Certain analyses can be thought of as "dimensioning independent." These analyses do not require the dimension, layering, or left/right information from the assembly representation described above. They need only the information shown on the graph – called a *liaison graph* – in Figure 5a that identifies distinct features on each part and the liaisons that connect them. The liaison graph is uniquely specified for any assembly; it is simply a graph representation of the liaisons assigned by the designer (we could think of it as an assembly's *liaison configuration*.) But, as shown in Figure 5b, a single liaison graph may represent many different assemblies. This is a useful attribute of the liaison graph, because we can perform the same analysis for an entire family, or "class", of assemblies.

3.2 Assembly Analysis

Assembly analysis using GapSpace consists of developing a graph representation of an assembly, and then traversing the assembly or liaison graph to establish the appropriate *Fits Conditions* for the assembly. Because assemblies with components fixed to one another with mating conditions are easy to analyze with traditional methods and software, we will focus on "floating assembly" examples where there are no mating conditions specified.

3.2.1 Traversing liaison graphs

As discussed in Section 3.1, liaison graphs are a spare form of assembly graphs, consisting of groups of part features joined by liaison arcs. All cycles on an assembly graph can also be represented on the corresponding liaison graph. These cycles can be described unambiguously by listing the liaison arcs belonging to the cycle in traverse order. The representation of a cycle on a liaison graph is shown in Figure 6. This figure shows a cycle that is in the direction of the liaisons which are traversed.

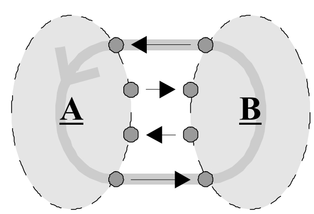


Figure 6:A cycle on a liaison graph.

3.2.2 Discerning assembly cycles

Once the assembly is represented as a graph, the assembly analysis can be performed by examining cycles on the graph. If we consider only cycles which follow the direction of liaison arcs, a non-negative sum of the liaison arc distances will correspond to a constraint condition on the part dimensions as thus: the sum of the signed distances around the cycle will be zero, and the requirement that the sum of the liaison distances be non-negative translates to the requirement that the sum of the signed dimension distances be non-positive.

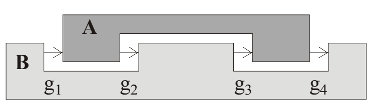
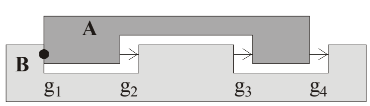
The three basic rules for determining valid assembly cycles are listed in Table 1. It should be noted that these rules apply to assemblies with an arbitrarily large number of parts, and those which have clearance conditions or both clearance and mating conditions.

Table 1: Some rules for valid assembly cycles.

1) Assembly cycles must follow the direction of the clearance condition liaison arcs (CCs).
2) No part may appear more than once in an assembly cycle.
3) If there is a mating condition (MC) between two parts in an assembly, any assembly cycle containing these two parts must contain this MC liaison. (*note: the direction of the MC liaison arc need not be followed.)*

The complete list of rules, and proof that they lead to the necessary and sufficient conditions for assembly, may be found in [2]. The simple examples shown in Table 2 illustrate the character of these rules and how mating conditions influence the analysis. Of course, the choice of dimensions for the assembly components will determine how these liaison constraints are applied. We call the liaison constraints imposed by the assembly cycles *Fits Conditions*, or FCs.

Table 2: Liaison constraints found from assembly cycles.

Assembly specification	Fits Conditions
"Floating" 	$g_1 + g_2 \geq 0$ $g_2 + g_3 \geq 0$ $g_3 + g_4 \geq 0$ $g_1 + g_4 \geq 0$
"Fixed" 	$g_1 + g_2 \geq 0$ $g_3 - g_1 \geq 0$ $g_1 + g_4 \geq 0$

The physical meaning of the liaison constraints can be thought of in the following way. Consider the constraint $g_1 + g_2 \geq 0$ for the floating assembly. The quantity $g_1 + g_2$ is equivalent to the size of the first "hole" in part B less the size of the first "pin" in part A. The other liaison constraints have analogous meanings. If all of these constraints are satisfied, the parts can be positioned so that they assemble without interference.

The fixed assembly has the constraint $g_3 - g_1 \geq 0$ that comes from Rule 3 in Table 1 (note that only clearance condition arc directions must be followed). This may be re-written as $g_3 \geq g_1$. If g_3 is greater than g_1, then if we slide part A to the left, g_1 will become zero before g_3. This is consistent with the mating condition requirement that A and B be in contact at their left most interface. We note here that if the parts are actually assembled with $g_1 = 0$, the fits conditions reduce to $g_2 \geq 0$, $g_3 \geq 0$, and $g_4 \geq 0$. This is the analysis we expect for an assembly where the relative motion between parts is removed. We keep g_1 in this analysis because we wish to be able to perform assembly analysis even when the parts are not in their nominally assembled state.

We now introduce GapSpace using the floating assembly from Table 2. We'll label the Fits Conditions FC_1-FC_4, and recall that assembly is possible when all of the $FC_i \geq 0$.

$$\begin{aligned} FC_1 &= g_1 + g_2 \geq 0, \quad FC_2 = g_2 + g_3 \geq 0, \\ FC_3 &= g_3 + g_4 \geq 0, \quad FC_4 = g_1 + g_4 \geq 0. \end{aligned} \tag{1}$$

Each of these fits conditions corresponds to the total free play, or gap, between pairs of mating part features. When the fits condition has a non-negative value, the features can fit together. In other words, *they assemble.*

We define a four-dimensional vector space (the GapSpace for this assembly) as the space spanned by the four individual gap values, which we initially assume to be independent. A subspace of GapSpace, called assembly space, is spanned by the four fits conditions. Note that because the fits conditions are linearly dependent, the space spanned by the four fits conditions is three dimensional[1].

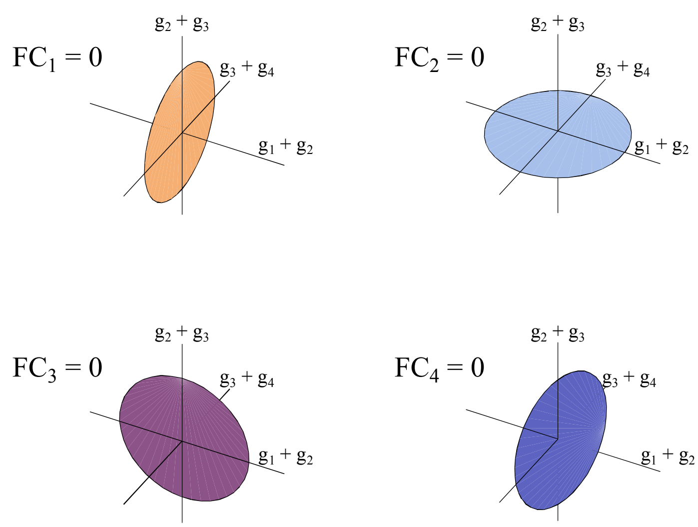


Figure 7: Fits conditions at equality.

The three gap pairs corresponding to FC_1, FC_2, and FC_3 are chosen as orthogonal basis vectors for the assembly space, while the fourth gap pair (g_1+g_4) is a linear combination of these three. The four fits conditions $(FC_i \geq 0)$ define planar

[1] While this paper makes extensive use of the hinge example which permits the visualization of a three-dimensional assembly space, the theory described herein may be applied to more complex analyses where the assembly space has more than three dimensions. The assembly space is no longer visualized easily for these cases, but the vector-space representation and the containment properties we show graphically for the hinge example still hold.

halfspaces in the assembly space, and we show their boundaries (FC_i = 0) in Figure 7. These halfspaces have infinite extent although they are shown as disks in the figure; for example, the plane for $FC_2 = 0$ corresponds to the *entire* "xy-plane."

If we consider the region in the assembly space where all of the fits conditions are satisfied, it is the intersection of the halfspaces shown in Figure 7. This intersection, which we call the assembly region, is shown in Figure 8. Because the individual halfspaces are infinite, this assembly region also is unbounded in the direction of increasing gap sizes (and FCs) away from the origin.

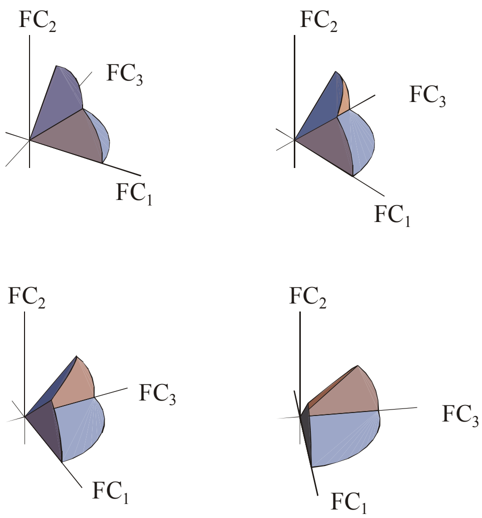


Figure 8:Four views of the assembly region.

In section 5 of this paper we show that the tolerances for assembly components can be modeled in the same space, and compared to this assembly region to perform assembly analyses.

3.2.3 Modeling relative motion

It is the nature of Fits Conditions that they are invariant under motion of the assembly components. Also, a Fits Condition value represents the total gap, or "free play" between a two pairs of features. In [2] we show how individual assembly cycles and sums of assembly cycles form relative motion walks (RMwalks) on the assembly graph. An RMwalk is the union of two non-self-intersecting traversals on the assembly graph: one from the first part to the second, and the other from the second part to the first. The walk itself may be self-intersecting, so long as the component traversals are not.

Figure 9a shows an RMwalk for the motion of part **A** relative to **C** on a liaison graph for the three part assembly shown in Figure 9b. This is not a valid assembly cycle, because it passes through part **B** more than once, violating Rule 2 in Table 1. If the sum of the gap sizes in this walk is the minimum of all RMwalks for parts **A** and **C**, then it will correspond to the maximum relative motion possible between parts **A** and **C**. In this example, the relative motion is constrained by the third part, **B**.

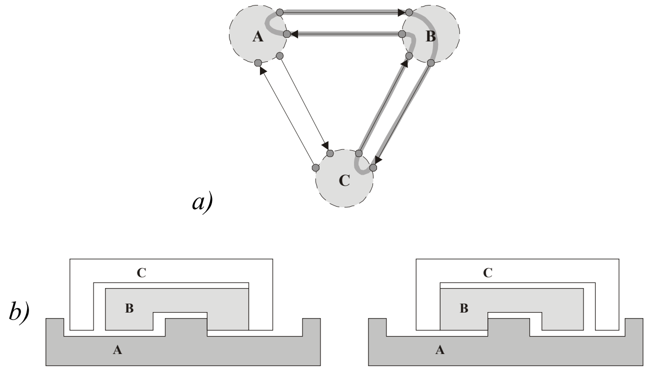


Figure 9: The motion between A and C is constrained by B.

Multiple RMwalks may correspond to each pair of parts in an assembly, and the RMwalk having the minimum value corresponds to the relative motion that can occur between the parts. In a two part floating assembly (no Mating Conditions), the RMwalks are identically the Fits Conditions, and the Fits Condition with the smallest value describes the possible relative motion between the parts.

4 ASSEMBLY METRICS AND CRITERIA

Because our model does not require the assembly components to be fixed, we can analyze the relative motion between components in the assembly. This gives rise to two measures of how well the assembly "goes together." The first is the size of the largest gap that is possible in the assembly, and the second is the relative motion that is possible between components. These metrics may be evaluated for a specific set of rigid components, or evaluated for the class of all in-tolerance parts.

A primary criterion for assemblies of rigid components is that the parts can fit together without interference. We call this "assemblebility." The other two criteria we use for assemblies are determined by placing numerical limits on the metrics described above. For instance, "the largest gap that is permissible between two components in the assembly is ***x***", or "the maximum relative motion between two parts in the assembly is ***y***." We call these criteria maximum possible gap (MPG) and maximum relative motion (MRM).

4.1 Assemblebility

We have already shown the assemblebility region for the hinge example in Figure 8. This criterion places a lower limit (of zero) on all four FC values. The assembly region is the intersection of the halfspaces for *all* FCs in the assembly space, and is unbounded for large gap sizes. The criterion that the parts assemble must be complemented by other criteria that provide an "upper" bound on the region in assembly space. Criteria that place a limit on the MRM and MPG functions accomplish this by placing upper bounds on one or more of the FC values.

4.2 Maximum Relative Motion (MRM)

The MRM criterion does not guarantee a bounded assembly region because it places an upper bound on only the *minimum* RMwalk in the assembly. Therefore, FCs that are not part of the minimum RMwalk can be arbitrarily large without affecting the satisfaction of the criterion. Figure 10a shows a simple assembly whose RMwalks are identically the FCs: $\mathrm{FC}_1 = \mathrm{g}_1 + \mathrm{g}_2$, $FC_2 = g_1 + g_3$. Because the MRM criterion depends on only the *minimum* FC value, leaving the other FC value unbounded, the specification of $\mathrm{MRM} < x$ results in an unbounded region in assembly space, shown in Figure 10c.

4.3 Maximum Possible Gap (MPG)

The MPG criterion places a limit on the maximum size of any of the clearance condition liaisons (CCs, or gaps) when the parts are assembled, resulting in a bounded region in assembly space. Because this criterion is in effect when the parts are assembled, no CC may have a negative value, and the region corresponding to this criterion must be a subset of the assembly region. Because all CCs must be non-negative, no CC may exceed the value of the FCs (positive sums of CC values) in which it appears. For the assembly shown in Figure 10a, $\mathrm{g}_1 \in \mathrm{FC}_1$, $\mathrm{g}_2 \in \mathrm{FC}_2$, and $\mathrm{g}_3 \in \mathrm{FC}_1$ and FC_2. The maximum value g_1 can achieve while the parts are assembled is FC_1; similarly g_2 cannot exceed FC_2 and g_3 cannot exceed the minimum of FC_1 and FC_2. Thus, the MPG for the assembly is equal to the maximum FC value and the region in assembly space described by MPG $\le x$ is simply the square described by $0 \le FC_1 \le x$ and $0 \le FC_2 \le x$, as shown in Figure 10d.

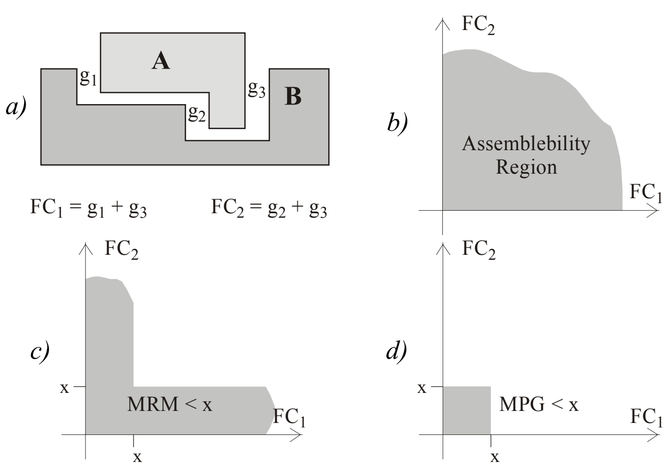


Figure 10:Assembly with 2 FCs, showing regions for assemblebility, MRM, and MPG

The hinge assembly provides a more interesting example of the MPG criterion because of the different ways in which the MPG can be constrained. First consider the relationship between the clearance condition values and the FCs:

$$\begin{aligned} &\mathrm{FC}_1 = \mathrm{g}_1 + \mathrm{g}_2, && \mathrm{FC}_2 = \mathrm{g}_2 + \mathrm{g}_3, \\ &\mathrm{FC}_3 = \mathrm{g}_3 + \mathrm{g}_4, && \mathrm{FC}_4 = \mathrm{g}_1 + \mathrm{g}_4. \end{aligned} \tag{2}$$

The membership of the gap values in the FCs is shown below:

$$\begin{aligned} &\mathrm{g}_1 \in \mathrm{FC}_1 \text{ and } \mathrm{FC}_4, && \mathrm{g}_2 \in \mathrm{FC}_1 \text{ and } \mathrm{FC}_2, \\ &\mathrm{g}_3 \in \mathrm{FC}_2 \text{ and } \mathrm{FC}_3, && \mathrm{g}_4 \in \mathrm{FC}_3 \text{ and } \mathrm{FC}_4. \end{aligned} \tag{3}$$

The maximum value that g_1 can achieve is the minimum of FC_1 and FC_4. So, to bound g_1 above by the value x, we require that:

$$(\mathrm{FC}_1 \le x) \vee (\mathrm{FC}_4 \le x) \tag{4}$$

A similar relationship exists for the other gap values. So, to bound the maximum gap value for the entire assembly by x we have the following equation:

$$\begin{aligned} &[(\mathrm{FC}_1 \le x) \vee (\mathrm{FC}_4 \le x)] \wedge [(\mathrm{FC}_1 \le x) \vee (\mathrm{FC}_2 \le x)] \wedge \\ &[(\mathrm{FC}_2 \le x) \vee (\mathrm{FC}_3 \le x)] \wedge [(\mathrm{FC}_3 \le x) \vee (\mathrm{FC}_4 \le x)] \end{aligned} \tag{5}$$

This may be reduced through logical manipulation to

$$\underbrace{[(FC_1 \le x) \wedge (FC_3 \le x)]}_{\textit{first half}} \vee \underbrace{[(FC_2 \le x) \wedge (FC_4 \le x)]}_{\textit{second half}} \text{, where } \mathrm{FC_i} \ge 0, \forall \mathrm{i}. \tag{6}$$

In order to represent this MPG region in GapSpace, we consider the two halves of Equation (6) separately. Using the information in the first half of the equation and the requirement that the parts are assembled, the following constraints describe a region in GapSpace:

$$\underbrace{(0 \le FC_1 \le x) \wedge (0 \le FC_2) \wedge (0 \le FC_3 \le x) \wedge (0 \le FC_4 \le x)}_{\textit{first half}} \tag{7}$$

The constraints on the FCs in Equation (7) result in the region shown in Figure 11a, while the region derived from the second half of Equation (6) is shown in Figure 11b.

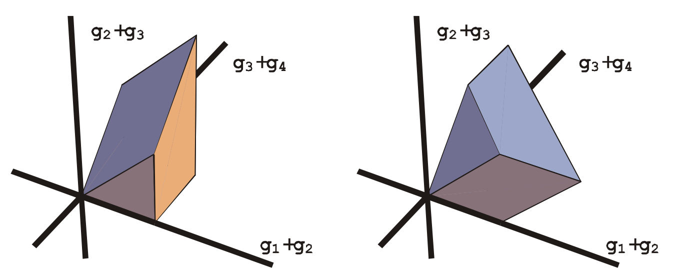


Figure 11: The regions described by the two halves of Equation (6).

Because the two halves of Equation (6) are joined by a logical "or", we can represent the entire MPG equation with the set union of the two regions shown in Figure 11; a point representing an assembly instance is either in the first region *or* the second region. This set union results in the region shown in Figure 12.

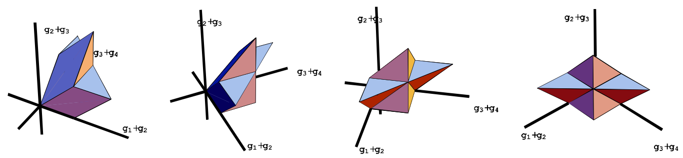


Figure 12: Four views of the (MPG ≤ *x*) region for the hinge assembly.

An interesting feature of the MPG criterion region described above is that although it is bounded, it is not convex. This faceted shape is peculiar to the logical structure of the MPG requirement. The table that follows summarizes the properties of the regions in assembly space that correspond to the different assembly criteria.

Table 3: Regions described by assembly criteria

	Criterion		
	Assemblebility	**MRM**	**MPG**
Bounded?	**Never** [1]	**Sometimes**	**Always** [2]
Convex?	**Always** [3]	**Sometimes**	**Sometimes**

Notes from Table 3:

1. The assemblebility region is the intersection of the unbounded GapSpace region (quadrant/ octant/ hyper-octant) described by $CC_j \geq 0$ and the lower-dimensional assembly space spanned by the FC_i. Since the CC are linearly independent, any linear combination of the CCs (such as the FCs) is also unbounded.
2. If the MPG region is unbounded, then some FC is unbounded and there must be an unbounded CC contributing to the FC. However, an unbounded CC will violate any finite MPG specification.
3. The assemblebility region is the intersection of planar halfspaces described by $FC_i \geq 0$. These halfspaces are convex, so their intersection is convex.

5 ANALYSIS OF TOLERANCED ASSEMBLIES

In this section we show how assemblies whose components have different tolerances can be compared with respect to a common assembly criterion. For our example, we will use the criteria that (1) the parts assemble and (2) the maximum possible gap (MPG) can be no more that one unit, for any pair of in-tolerance parts in any assembled position.

We will use the hinge assembly described in the earlier sections with three different methods of placing dimensions and tolerances on the part. In each case, we will develop a tolerance region in the GapSpace that represents all possible combinations of in-tolerance components. If the tolerance region is contained in region described by the MPG criterion, the tolerance specification will satisfy the criterion.

In this analysis we infer the range of possible FC values from the part dimensions. In Figure 13, the value of FC_1 $(g_1 + g_2)$ is determined by the left-most dimensions on the two parts. The possible range of values for FC_1, given that the individual parts are in tolerance, is simply the minimum difference in the dimensions $(10 - 9\frac{2}{3} = \frac{1}{3})$ to the maximum difference $(10\frac{1}{3} - 9\frac{1}{3} = 1)$. Note that the value of FC_1 is independent of the part location, using the convention that overlapping parts result in negative gap values.

5.1 Chain dimensioning

In this case, we apply the dimensions (with tolerances) to adjacent pairs of features on each of the components. The tolerances are balanced, in that the same spread of values is permitted for each dimension, and all dimensions are assumed to be independent. If we consider the possible values that the FC_1 $(g_1 + g_2)$ can take on, we see that this range is $[\frac{1}{3}, 1]$. The same range is possible for FC_3, while FC_2 can take on the range $[0, \frac{2}{3}]$. Figure 16a shows the tolerance region of all points satisfying these ranges for the FCs. This tolerance region is just contained within the MPG criterion region.

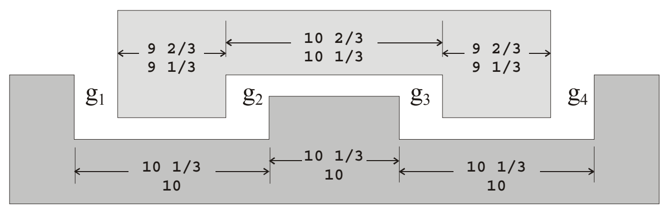


Figure 13: "Chain" dimensioning for the hinge components.

5.2 Baseline dimensioning

In this case, the dimensions are applied relating each feature of the components back to the left-most feature of that component. This method has value in that the accumulation of variability between opposite ends of the part is minimized. However, we see that the ranges allowed for the individual dimensions is less in than the previous example. While the dimensions shown on the drawing are assumed to be independent, the FCs that we plot in Figure 16b are not. For example, both FC_1 and FC_2 are dependent on the leftmost dimension of the parts. This dependence is shown by the slope of the "top" and "bottom" surfaces of the tolerance region, exhibiting a correlation between FC_1 (g_1+g_2) and FC_2 (g_2+g_3).

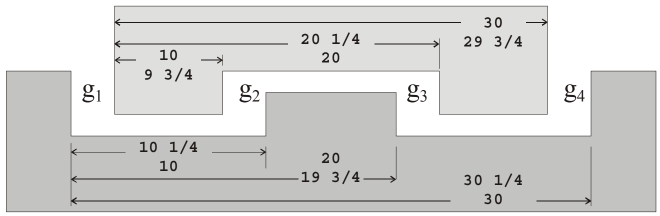


Figure 14: "Baseline" dimensioning for the hinge components.

5.3 Geometric Dimensioning

The components shown in Figure 15 have their dimensions and tolerances specified in a one-dimensional application of the ASME dimensioning and tolerancing standard [17]. The interesting result of this specification is the dependence in the dimensions allowed by the maximum material condition modifier. Thus, the relative location of the feature pairs on the end of each part is allowed to vary more and more as the feature pairs get smaller (upper part) or larger (lower part). This dependency is apparent in Figure 16c, as this is the only tolerance region which is not described by 3 pairs of parallel faces, corresponding to the 3 pairs of independent dimensions.

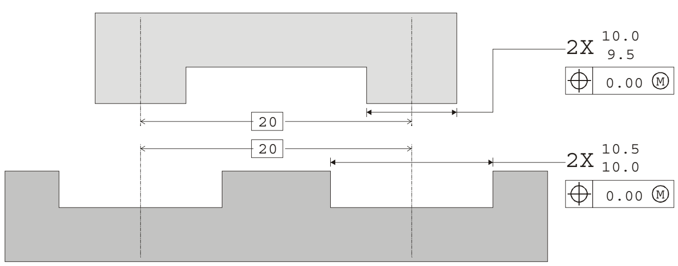


Figure 15: Geometric tolerances for the hinge components.

5.4 Comparison

In Figure 16 we show each of the tolerance regions with respect to the MPG criterion from Section 4. In each case, the tolerance region "just fits" within the region described by the MPG criterion. This means that loosening any of the tolerances would result in a larger tolerance region that is not entirely contained in the criterion region, and hence does not satisfy the criterion. If we use the chain dimensioning scheme to tolerance our parts, we can specify looser tolerances that if we use the baseline scheme: ***this almost certainly means that the parts can be made less expensively!*** The geometric tolerancing scheme appears to be better still, as these tolerances will accept more parts that either of the other two methods. The difficulty is now in choosing a manufacturing process that can best meet the tolerances specified.

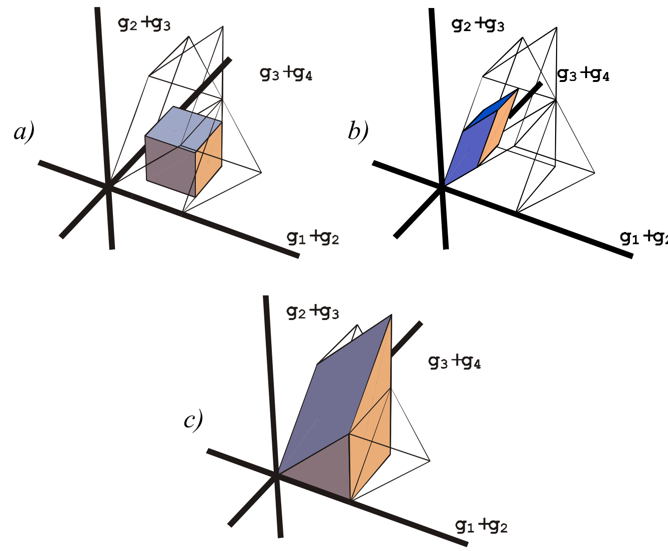


Figure 16: Different tolerance regions that are just contained within the (wireframe) MPG region.

A final note regarding Figure 16: the selection of FC_1, FC_2, and FC_3 as the basis for the plot coordinate systems is arbitrary, and does not effect the results of the tolerance analysis. Figure 17 shows the same tolerance regions and MPG criterion region with a different basis. The coordinate axes for these plots are FC_1, FC_4, and FC_2-FC_1 (g_3-g_1). This basis corresponds to the baseline dimensioning scheme, which is clear from Figure 17*b)*, where the tolerance region is described by a cube. These regions are unchanged in their relationship to the MPG criterion and to each other; they have just been distorted by the change of coordinate system.

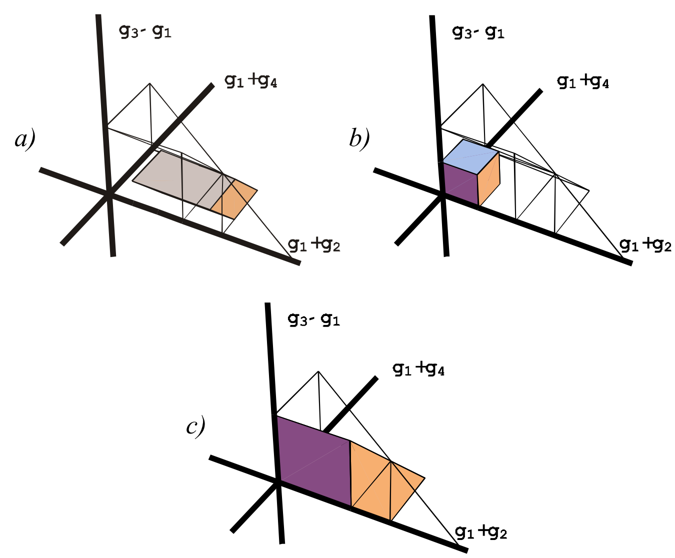


Figure 17: MPG region and tolerance regions with a different coordinate system basis.

6 SUMMARY

In this paper we have shown how the GapSpace model can be used to represent different assembly criteria and toleranced assemblies in a common environment. We reviewed briefly the structure of GapSpace, and then showed how three fundamental criteria for describing assemblies are represented in this space. These criteria are represented as regions in the assembly space, called the *assembly criterion regions*. The toleranced dimensions of assembly components describe a differently-shaped region in the assembly space, called the *tolerance region*. When the tolerance region is contained (in a set containment sense) in an assembly criterion region, we have specified a set of part tolerances that guarantees the satisfaction of the assembly criterion.

The GapSpace model supports statistical analysis, so probabilistic conformance to various criteria can also be assessed. We have developed software to extract model information from a commercial modeling package and perform assembly analysis external to the modeler. We are currently extending this work to higher-dimensional analysis (2- and 3-D assembly models).

ACKNOWLEDGMENTS

This research was supported in part by a DOE pre-doctoral fellowship in Integrated Manufacturing. I am grateful to my colleagues who encouraged me to pursue this work and to Zhihua Zou, whose continuing work in this area provides additional insights about the nature of assemblies.

REFERENCES

[1] Parratt, S. W., "A Theory of One-Dimensional Tolerancing for Assembly", Ph.D. Thesis, Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY, 1994.

[2] Morse, E. P., "Models, Representations, and Analyses of Toleranced One-Dimensional Assemblies", Ph.D. Thesis, Sibley School of Mechanical and Aerospace Engineering, Cornell University, Ithaca, NY, 1999.

[3] Morse, E. P., "Introduction to GapSpace: A Representation of Assemblies for One-Dimensional Analysis", in review: *Journal of Mechanical Design*, October 2000.

[4] Voelcker, H. B., "Remarks on the Essential Elements of Tolerancing Schemes", in **Geometric Design Tolerancing: Theories, Standards, and Applications**, H. A. ElMaraghy, Ed., Chapman and Hall, London, pp. 119-121, 1998.

[5] Requicha, A. A. G., "Part and assembly description languages I: Dimensioning and Tolerancing", technical memorandum TM-19, Production Automation Project, University of Rochester, May 1977.

[6] Turner, J. U., Wozny, M. J., "A mathematical theory of tolerances", **Geometric Modeling for CAD Applications**, M. J. Wozny, H. W. McLaughlin, and J. L. Encarnaçao, Eds., Elsevier Science Publishers B. V., North-Holland, pp. 163-187, 1988.

[7] Dorminey, D. "An Exploratory Study of Tolerance Classes for Interchangeable Assembly in One Dimension", M.S. Thesis, Cornell University, Ithaca, NY, 1999.

[8] Bjørke, Ø., Computer Aided Tolerancing, ASME Press, New York, NY, 1989.

[9] Juster, N. P., Dew, P. M., and de Pennington, A., "Automating linear tolerance analysis across assemblies", *Journal of Mechanical Design*, **114**, no. 1, pp. 174-179, March 1992.

[10] Greenwood, W. H., and Chase, K. W., "A new tolerance analysis method for designers and manufacturers", *Journal of Engineering for Industry*, **109**, pp. 112-116, May 1987.

[11] Chase, K. W., Gao, J., and Magleby, S. P., "General 2-D tolerance analysis of mechanical assemblies with small kinematic adjustments", Journal of Design and Manufacturing, **5**, no. 4, pp. 263-274, December 1995.

[12] Gao, J., "Nonlinear Tolerance Analysis of Mechanical Assemblies", Ph.D. Thesis, Brigham Young University, Provo, UT, 1993.

[13] Clement, A., Desrochers, A., and Riviere, A., "Theory and practice of 3-D tolerancing for assembly", *Proc. C. I. R. P. International Working Seminar on Computer-Aided Tolerancing*, Penn State University, University Park, PA, pp. 25-55, May 1991.

[14] Fleming, A. D., "Analysis of Uncertainties and Geometric Tolerances in Assemblies of Parts", Ph.D. Thesis, University of Edinburgh, Edinburgh, UK, 1987.

[15] Mullins, S. H., and Anderson, D. C., "Automatic identification of geometric constraints in mechanical assemblies", Computer-Aided Design, **30**, no. 9, pp. 715-726, August, 1998.

[16] Requicha, A. A. G., and Tilove, R. B., "Mathematical foundations of constructive solid geometry: General topology of closed regular sets", technical memorandum TM-27a, Production Automation Project, University of Rochester, June 1978.

[17] **Dimensioning and Tolerancing**, American Society of Mechanical Engineers, Standard Y14.5M - 1994 (reaffirmed 1999).

