# Automatic Generation of Tolerance Chains from Mating Relations Represented in Assembly Models

**N. Wang**

**T. M. Ozsoy**

Department of Mechanical Engineering and Mechanics,
Lehigh University,
Bethlehem, PA 18015

*This paper presents an algorithm for generating tolerance chains from the mating relations between components of assemblies. The algorithm is developed upon a feature-based assembly modeling strategy that represents each component in close relation to its mating features, dimensions, and tolerances. The mating relations within an assembly are described by a mating graph. Tolerance chains together with their dimensions and tolerances are generated automatically by searching through a mating graph for matching mating features. A prototype program package based on the presented algorithm has been developed, and several examples of various complexity have been tested with success.*

## Introduction

The purpose of tolerance analysis is to determine an optimal tolerance assignment that can guarantee the functionality of the assembly, and meanwhile minimize the cost of production. Computer aided tolerance analysis is considered to be an important component of the Computer Integrated Manufacturing (CIM) technology. A number of software packages have been developed to conduct tolerance analysis (Bjorke, 1978; Parkinson, 1985; and Lehtihet and Dindelli, 1989), but very few have the ability to directly use the information stored in a CAD database.

The basic information needed by a tolerance analysis package is often in the form of a tolerance chain that describes the dimensional interrelations between member components of an assembly. Currently tolerance chains are usually specified by the user. The task may turn out to be difficult and prone to mistakes especially when the assemblies in question are complicated. To automate this process and make it more reliable, this paper presents an algorithm that can automatically derive the tolerance chain of a given assembly by using the information available in a feature-based assembly data structure.

## Tolerance Technology

Tolerance analysis and tolerance synthesis are two major categories of tolerance analysis techniques. Tolerance analysis investigates the effects of individual dimensions on a particular one, named sum dimensions by Bjorke (1978). Conversely, tolerance synthesis determines individual dimensions according to a specified sum dimension. Both methods, however, require finding the relations between the sum dimension and the individual dimensions, which is known as the fundamental equation and described by the tolerance chain.

According to Bjorke's definition (1978), sum dimensions are those dimensions that affect the function of an assembly more than any other dimensions. The influence of each individual dimension to the sum dimension is statistically summed up along a specified direction—the sum direction that is a 3D vector in the space. A sum dimension is between either two stationary parts, or one stationary and one rotationary part, or two rotationary parts. Different methods are used in the summing procedures for the three different types of sum dimensions.

A tolerance chain can be represented in the form of a graph in which nodes represent the mating features, and arcs, known as the chain links, stand for the tolerances of the dimensions between the mating features they connect. Tolerance chains are classified as either simple related chains or interrelated ones, depending on whether the correspondent graph contains a single loop or multiple ones (even netted). Chain links can also be classified as spans and gaps. Spans are further divided into line vector spans, plane vector spans and space spans. Gaps are grouped into line vector gaps, including clearance, transition and interference, and plane vector gaps (Bjorke, 1978). The mathematical interpretation of a tolerance chain leads to a fundamental equation that is of critical importance in both tolerance analysis and tolerance synthesis.

Illustrated in Fig. 1(*a*) and (*b*) is a wheel mounting assembly and its tolerance chain. It is an example of a design with interrelated tolerance chains. $X_{\Sigma_1}$ and $X_{\Sigma_2}$ are chosen to be the sum dimensions because they directly affect the functionality of the assembly, and their tolerances have to be satisfied simultaneously. The mathematical interpretation of the tolerance chain yields the following fundamental equations:

$$X_{\Sigma_1}=f_1(X_i)=X_2-X_4$$
$$X_{\Sigma_2}=f_2(X_i)=-X_1-X_2-X_3+X_5 \quad (1)$$

## Representational Scheme

Assembly modeling deals with the inter-relationships between assembled machine parts (components), rather than the

Contributed by the Design Automation Committee for publication in the JOURNAL OF MECHANICAL DESIGN. Manuscript received Oct. 1990; revised July 1991. Associate Technical Editor: B. Ravani.

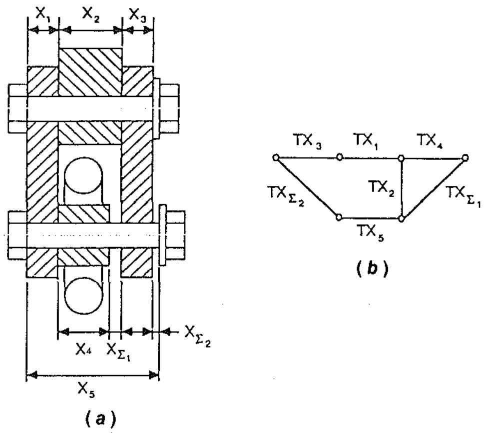


**Fig. 1** (*a*) **The wheel mounting assembly (*b*) the tolerance chain (Bjorke, 1978)**

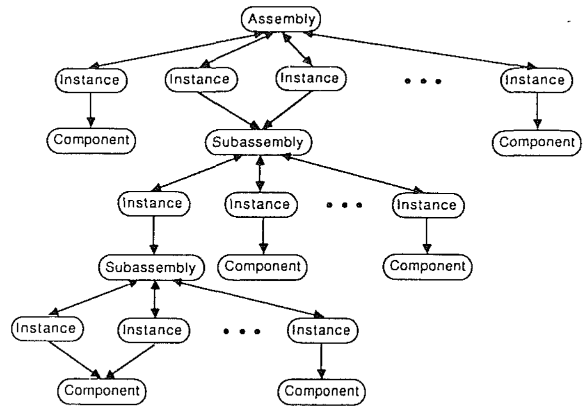


**Fig. 2 The assembly graph**

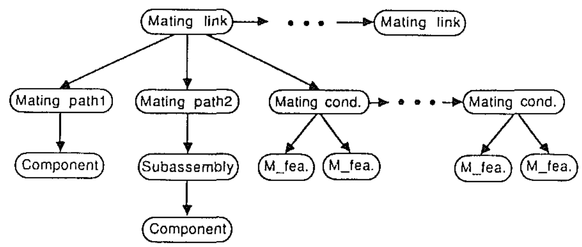


**Fig. 3 Mating links**

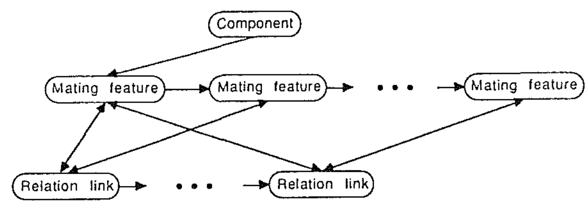


**Fig. 4 The relations between components, mating features, and relation links**

detailed shapes of each part. However, in order to support applications such as tolerance analysis of assemblies, certain geometric features of the individual parts and the corresponding dimensional and variational information must be represented in the model data base. A feature-based assembly modeling strategy is therefore developed (Wang, 1990).

**Representation of Mating Relations in Assemblies.** An assembly graph (Fig. 2) is designed to represent the assembly, its subassemblies, and components in a graph structure as the topmost (root) node, intermediate nodes, and terminal nodes, respectively. Since a component or a subassembly should be allowed to occur more than once at different locations with different orientations in an assembly, the concept of instance is introduced.

The connectivity information between the elements of an assembly is made available through the instances instead of through the components or subasssemblies. Each instance has a pointer to a set of mating links that are introduced by the authors to record the mating information. Whenever a component or a subassembly is assembled, an instance is created automatically. The first instance attached to an assembly or a subassembly does not have mating links. For the subsequent instances, mating links are created and linked together according to the user specified mating conditions. The position and orientation of an instance is derived from the mating conditions carried by the mating links of that instance.

A mating link stores the mating relations between a pair of mating entities which could be either two components (C-C mating), two subassemblies (S-S mating), or one component and one subassembly (C-S mating). For S-S and C-S mating, since the mating eventually happens between two components, the "mating path" is introduced to make the mating unique, which may not be the case otherwise due to the use of instances. A mating path that is associated with a subassembly traces the assembly graph from that subassembly all the way down to the member component where the mating actually exists. A mating path that is associated with a component has only one element, pointing directly to the component (Fig. 3).

In the case of S-S and C-S mating, there may be more than one pair of components or subassemblies mating each other, thus a set of mating links is needed. The detailed mating information about where and how the mating happens is provided by mating conditions and mating features. Since two components may mate each other at more than one contact, each mating link may also have more than one mating condition node.

Currently, only three mating conditions are implemented, which are *against*, *parallel*, and *fit*. A mating condition "against" constrains two planar faces in such a way that the two faces will touch each other with their normals pointing at the opposite directions. A mating condition "parallel" also constrains two planar faces. The two constrained faces will be parallel to each other at a specified distance and with their normals pointing at the same direction. A mating condition "fit" constrains two cylindrical faces so that their center axes are aligned.

**Representation of Components and Mating Features.** Differed from the conventional assembly modeling, components are represented with their mating features (Fig. 4). Mating features contain the specific geometrical information referred to by mating conditions. For instance, if the mating condition is against, the two mating features will be two planar faces, each on one of the two mating components; if the mating condition is fit, the two mating features will be two cylindrical faces, etc.

Mating features are defined by the user whenever a component is created. So far as the tolerance chain generation is concerned, a component and its mating features can be represented either with or without an underlying object data structure. A component can point to an existing object and its mating features can be created by selecting the corresponding faces of the object. If there is no object data structure available, a component can be represented as a "dummy" without detailed shape descriptions. Such a component is created by the user specifying its affiliated mating features in which the key geometrical information will be explicitly stored, e.g., a point and a normal vector stored for a planar face, an axis, and a diameter for a cylindrical face, etc. This second method is currently being used in the presented approach, though a feature-based object data structure has become available (Wang, 1990).

**Representation of Dimensions and Tolerances.** Other important pieces of geometrical information are the dimensions and tolerances between the mating features that belong to the same component. This information is either specified by the user when the components are created without referring to the underlying objects, or derived from the feature-based object data structure directly. Once it has been defined or derived, it will be stored with the relation links (Fig. 4). For example, if a component has two mating features that are two parallel through holes, the positional dimension between them and the corresponding tolerances will be stored in a relation link which points to the two mating features.

## Generation of Tolerance Chains

A tolerance chain describes the relations between a user specified sum dimension and the individual dimensions of a given assembly. These relations are implicitly embedded in the mating information provided by the assembly graph. In order to generate tolerance chains of an assembly automatically, the mating information carried by mating links is reorganized into a mating graph. Starting from one end of a sum dimension, chain links in a tolerance chain are located one at a time by searching through the mating graph and checking the mating features, until the other end of the sum dimension is reached. The algorithm is outlined below:

```
Procedure: Generate_tolerance_chain
  Find_mating_graph (Assembly-graph, Mating-graph)
  For each sum dimension in the assembly Do
    Specify the sum dimension and the associated sum
    direction
    Find_chain_links (Assembly-graph, Mating-graph, Tolerance-chain)
    Find_fundamental_equation (Tolerance-chain)
  Enddo
```

A mating graph is a graphical representation of mating relations between all components of an assembly. Each node of a mating graph stands for a mating component, and each arc stands for the mating relations between two mating components, which are described in a corresponding mating link. By searching through the assembly graph of an assembly, a unique mating graph is derived from the mating links. A depth-first searching proceeds from the top most node of the assembly graph. The algorithm is elaborated as follows in a pseudo language.

```
Procedure: Find_mating_graph (Assembly-graph, Mating-graph)
  Get the first Instance from the top most node of the
  Assembly-graph
  Create_mating_graph (Assembly-graph, Instance, Stack, Mating-graph)

Procedure: Create_mating_graph (Assembly-graph, Instance, Stack, Mating-graph)
  For each mating link in Instance Do
    For each mating path in the mating link Do
    If the mating component is not in the mating graph
    Then
      Add a new node in the mating graph
    Endif
  Enddo
  Insert a new arc into the mating graph
Enddo
If Instance is a subassembly Then
  Push_stack (Instance, Stack)
  Get the first Instance of the subassembly
  Create_mating_graph (Assembly-graph, Instance, Stack)
  Pop_stack (Stack)
Endif
If Instance has next pointer Then
  Get the next Instance
  Create_mating_graph (Assembly-graph, Instance, Stack, Mating graph)
Endif
```

Since tolerance chains are sum dimension dependent, different sum dimensions will lead to different tolerance chains even for the same assembly. To generate a tolerance chain, the sum dimension and its direction have to be specified first. In the present work, a sum dimension is represented by two mating features on two components named "sum components" by the authors. A sum direction is a 3D vector along which the weighted contributions from individual dimensions are summed up.

Each chain link in a tolerance chain represents a dimensional relation between two mating features of a component, thus it contains pointers to a component and two mating features. To generate a tolerance chain, one of the two sum components will be chosen to be the current component, and added into the chain as the first chain link. By searching through the mating graph, the components mating the current component can be identified. Starting with the first one, the associated mating relations, i.e., a set of mating conditions, are retrieved through the corresponding arc in the mating graph. For each pair of mating features defined by the mating conditions, the mating is checked to see whether it affects the sum dimension along the sum direction. If it does, a new chain link will be added into the chain, and the current component will be updated. If it does not, the next pair of mating features will be checked. The above procedure is repeated until the whole tolerance chain is generated completely.

The two criteria used to check whether a mating has influence on a given sum dimension are defined below.

- IF two mating features are planar faces and their normals are not perpendicular to the sum direction,
  THEN the mating specified by these two mating features has influence on the sum dimension.
- IF two mating features are cylindrical faces and their axes are not parallel to the sum direction,
  THEN the mating specified by these two mating features has influence on the sum dimension.

The algorithm presented above is elaborated as follows:

```
Procedure: Find_chain_link (Assembly-graph, Mating-graph, Tolerance-chain)
  Add the first sum component and the mating feature of
  the sum dimension into the tolerance chain as the first
  chain link
  Locate the first sum component in Mating-graph
Start:
  Get Current-component from the first arc of current node
  in Mating-graph
  For Current-component Do
    Push_stack (Current-component, Stack)
    If Current-component is the second sum component
    Then
    Add the second sum component and the mating feature
    of the sum dimension into the tolerance chain
    Finish one loop of a simple tolerance chain or
    interrelated one
  Else if Current-component is one of the chain links Then
    Finish one loop of an interrelated tolerance chain
```

```
Else
  For each mating condition referred by current arc in
  Mating-graph Do
    Get mating features from the mating condition
    If the mating has the influence on the sum dimension
    Then
      Add Current-component and the mating features
      into the tolerance chain as a new chain link
      Goto Start:
    Endif
  Enddo
Endif
If there is no arc left in current node of Mating-graph
Then
  If Stack is empty Then
    Stop the procedure
  Else Pop_stack (new Current-component, Stack)
  Endif
Else
  Get a new Current-component from the next arc of
  current node in the Mating-graph
Endif
Enddo
```

A fundamental equation is a mathematical interpretation of a tolerance chain. It contains the type (either span or gap), the scaling factor (the weight), and most importantly the dimension and tolerances of each chain link. Mating features of each chain link are the key information used to classify the chain link, to calculate the scaling factor, and to retrieve the dimension and tolerances. The procedure used to derive a fundamental equation is listed below:

```
Procedure: Find_fundamental_equation (Tolerance-
chain)
  For each chain link in Tolerance-chain Do
    Get mating features from the chain link
    Classify the chain link
    Calculate the scaling factor
    Get the dimension and tolerances either from the
    relation link or from the feature-based object data
    structure
  Add a new item in the fundamental equation
Enddo
```

A chain link will be a span if its mating features are planar faces. If the mating features are cylindrical faces, the corresponding chain link will be a gap. A scaling factor is the weight assigned to a term in a fundamental equation. It represents the amount of the influence of an individual dimension on a sum dimension. In the simplest case, which is also the most common case, the direction of an individual dimension in a tolerance chain is parallel to the sum direction, and, therefore, the scaling factor will be 1 or $-1$.

## Case Study

A prototype program package based on the above scheme has been developed, and several assemblies with various complexity have been tested successfully. As an example, a wheel mounting assembly (Fig. 1) is modeled and its tolerance chain is generated.

The wheel mounting assembly is composed of total eight components and its assembly graph is displayed in Fig. 5 where "Plate," "Bushing," "Wheel_mount," "Bolt_1," "Bolt 2," and "Bolt_head" are components. Each component has several mating features (Fig. 6).

The mating features of each component and their relations are summarized in Table 1.

The mating graph (Fig. 7) of the wheel mounting is derived from its assembly graph.

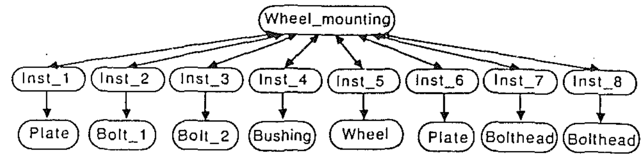


**Fig. 5 The assembly graph of the wheel mounting**

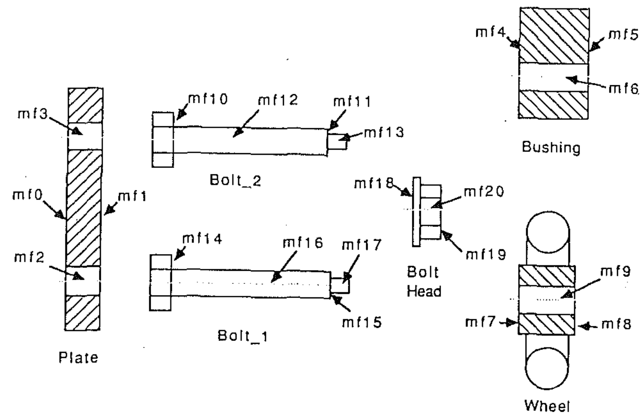


**Fig. 6 Components and their mating features of the wheel mounting assembly**

**Table 1 Component/mating feature, mating feature/mating feature relations**

| Components | Mating features | Relation between mating features |
|---|---|---|
| Plate | mf0,mf1,mf2,mf3 | mf0 ⟷ mf1, mf2 ⟷ mf3 |
| Wheel_mount | mf4,mf5,mf6 | mf4 ⟷ mf5 |
| Bushing | mf7,mf8,mf9 | mf7 ⟷ mf8 |
| Bolt_2 | mf10,mf11,mf12,mf13 | mf10 ⟷ mf11 |
| Bolt_1 | mf14,mf15,mf16,mf17 | mf14 ⟷ mf15 |
| Bolt_head | mf18,mf19,mf20 | mf18 ⟷ mf19 |

There are two sum dimensions in the wheel mounting assembly, which are the clearance gap between the wheel mounting and the right plate, and the gap between the right plate and the bolt head of the bottom bolt. The tolerances of these two sum dimensions are to be satisfied simultaneously. The corresponding sum components are "Plate" and "Wheel-mount," and "Plate" and "Bolt_head," respectively. The sum directions are all along the horizontal axis. The resultant tolerance chain has a netted loop, and its chain links are illustrated in Fig. 8.

The fundamental Eqs. (1) are derived from the tolerance chain (Fig. 8) and can be passed to a tolerance analysis and synthesis module (Treacy, 1988) through a generic formatted data file.

## Concluding Remarks

This paper presents an algorithm for automatic generation of tolerance chains by using mating relations represented in a feature-based assembly data structure proposed by the authors. Such an automation eliminates the need for human intervention to identify which dimensions in an assembly affect a given sum dimension, or how they are related. Tolerance chains are generated by searching through the mating graphs that are previously derived from mating links. This procedure is necessary because only the mating graph represents the global mating relations of an assembly, which is otherwise fragmental and implicit.

The assembly modeling strategy presented in this paper has the following advantages:

The natural structure of an assembly is retained in the as-

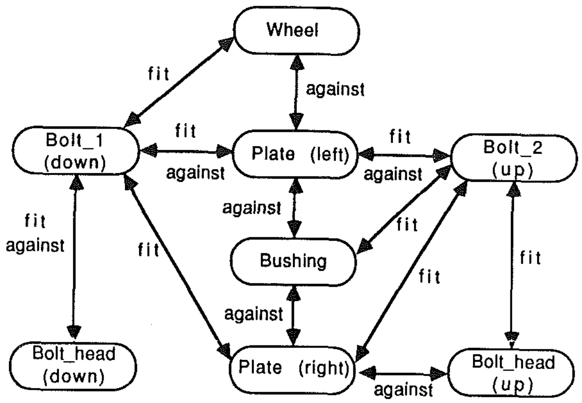


Fig. 7 The mating graph of the wheel mounting

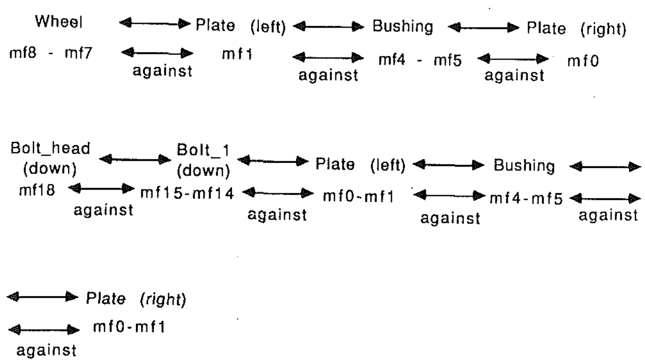


Fig. 8 The tolerance chain links of the wheel mounting

sembly graph, which cannot be explicitly described by a virtual-linked assembly structure proposed by Lee and Gossard (1985).

The use of instances for both components and subassemblies has been achieved by using mating paths to provide additional mating information so that any potential ambiguity can be avoided.

The assembly data structure is designed in such a way that it can either be stand-alone, thus being advantageous because it is more portable and flexible, or interfaced with a feature-based object data structure, which has already been implemented in the prototype package, for accessing the information about mating features and their dimensional and variational relations.

## Acknowledgments

This material is based on the work supported by the Digital Equipment Corporation's External Research Program.

## References

Bjorke, O., 1978, *Computer-Aided Tolerancing*, Tapir publisher, Trondheim, Norway.

Lee, K., and Gossard, D. C., 1985, "A Hierarchical Data Structure for Representing Assemblies: Part 1," *Computer Aided Design*, Vol. 17, No. 1, pp. 15-19.

Lehtihet, E. A., and Dinelli, B. A., 1989, "TOLCON: Microcomputer-Based Module for Simulation of Tolerances," *Manufacturing Review*, Vol. 2, No. 3, pp. 179-188.3.

Parkinson, D. B., 1985, "Assessment and Optimization of Dimensional Tolerances," *Computer Aided Design*, Vol. 17, No. 4, pp. 191-198.

Treacy, P. J., 1988, "A Tolerance Analysis and Synthesis Module for Relational Geometric Modeling Environment Interface," M.S. Thesis, Lehigh University.

Wang, N., 1990, "Feature-Based Geometric Modeling Toward the Integration of Design and Manufacturing," Ph.D. Dissertation, Lehigh University.