

Research Article

A Method for Designing Assembly Tolerance Networks of Mechanical Assemblies

**Yi Zhang,^{1,2} Zongbin Li,^{1,2} Jianmin Gao,^{1,2} Jun Hong,^{1,2}
Francesco Vilecco,³ and Yunlong Li^{1,2}**

¹ School of Mechanical Engineering, Xi'an Jiaotong University, Shaanxi 710049, China

² State Key Laboratory for Manufacturing Systems Engineering of China, Shaanxi 710049, China

³ Department of Industrial Engineering, University of Salerno, Via Ponte Don Melillo,
84084 Fisciano, Italy

Correspondence should be addressed to Yi Zhang, zhangyi.jd@stu.xjtu.edu.cn

Received 16 May 2011; Accepted 16 July 2011

Academic Editor: Carlo Cattani

Copyright © 2012 Yi Zhang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

When designing mechanical assemblies, assembly tolerance design is an important issue which must be seriously considered by designers. Assembly tolerances reflect functional requirements of assembling, which can be used to control assembling qualities and production costs. This paper proposes a new method for designing assembly tolerance networks of mechanical assemblies. The method establishes the assembly structure tree model of an assembly based on its product structure tree model. On this basis, assembly information model and assembly relation model are set up based on polychromatic sets (PS) theory. According to the two models, the systems of location relation equations and interference relation equations are established. Then, using methods of topologically related surfaces (TTRS) theory and variational geometric constraints (VGC) theory, three VGC reasoning matrices are constructed. According to corresponding relations between VGCs and assembly tolerance types, the reasoning matrices of tolerance types are also established by using contour matrices of PS. Finally, an exemplary product is used to construct its assembly tolerance networks and meanwhile to verify the feasibility and effectiveness of the proposed method.

1. Introduction

Assembly tolerance design is one of the research hotspots in the field of computer-aided tolerancing (CAT). Assembly constraints are essentially constraints between assembly feature surfaces of parts. Assembly tolerance is a vector constraint. Its orientation, type, and value are mainly determined by the assembly functional requirements of mechanical assemblies. Assembly tolerances are crucial for many activities in the product's life cycle, which not only affects assembly qualities but also determines manufacturing costs. Automatic generation of assembly tolerance networks can greatly reduce design complexity and improve design

quality. Meanwhile, optimization design of assembly tolerances can remarkably reduce manufacturing cost of mechanical assemblies. The design of assembly tolerance networks is a complex multiscale problem. It involves associations between the multiple scales, such as the assembly functional requirement, part positioning, datum reference frame, assembly sequence, assembly feature, and tolerance specification. The essences of many important issues are multiscale problems in medicine, physics, computers, chemistry, materials science, robotics, and other disciplines. Multiscale modeling and operations are widely used in these research fields. Kou et al. [1] used multiscale time schemes for simulating two-phase flow in fractured porous media. The method divides the total time interval into four temporal levels, which can take a large time-step size to achieve stable solutions. Picard et al. [2] introduced a new approach to model realistically sounding objects for animated real-time virtual environments. The number of hexahedral finite elements is only loosely determined by the geometry of the sounding object. Therefore, the approach can realize a multiscale solution. In [3], a multiscale time-frequency representation and the Stockwell transform were used to generalize the time-varying spectra and coherence, which makes the Stockwell transform an effective approach to investigate the characteristics of the spectrum and the interaction of locally stationary time series. The multiscale curvelet shrinkage and the total variation function are combined together, by which a new variational image model is established to compute an initial estimated image [4]. Bakhoun and Toma [5] established a higher-order differential equation to model multiscale phenomena and explain multiscale threshold transitions. Different delayed pulses and multiscale behaviour can be noticed when the order of the equation is higher. According to the multiscale continuous wavelet transform, Djilas et al. [6] presented an algorithm to separate activity of primary and secondary muscle spindle afferents. In [7], Chen and Beghdadi proposed a new algorithm for natural enhancement of color image by using the multiscale Retinex model, which can improve the luminance and chrominance contrast of image. Chen et al. [8] proposed a differential geometry-based multiscale framework to handle complex biological systems, which can deal with microscopic fluid dynamics, microscopic molecular dynamics, and surface dynamics in a unified framework. In [9], a multiscale linear system of algebraic equations is established to determine material motion and gradient velocity, which is adapted to the situation of tagging MRI data. M. N. Nounou and H. N. Nounou [10] developed a multiscale nonlinear modeling algorithm based on multiscale wavelet-based representation of data. The approach can enhance the estimation accuracy of the linear-in-the-parameters nonlinear model. By applying the same method in [10], Nounou and Nounou [11] presented a multiscale latent variable regression modeling algorithm to improve the prediction accuracy of some of the models, such as Principal Component Regression and Partial Least Squares. Lucia [12] developed a new multiscale methodology for cubic equations of state, which constraints the attraction parameter by using the Gibbs-Helmholtz equations.

In recent years, research on assembly tolerance design has made plentiful and substantial achievements. Based on TTRS theory, Clement et al. [13, 14] proposed a method to model and represent dimensional and geometric tolerances in CAT systems. Hoffmann [15] established a new model in three-dimensional space, in which geometric graph is regarded as a set of vector dots and tolerances are described by using tolerance functions in which point vectors are taken as parameters. Desrochers and Clément [16] provided a representation method of tolerance information based on TTRS theory, which is independent of modeling systems. In one-dimensional space, Wang and Ozsoy [17] established a model for generating tolerance chains according to mating relations between components of assemblies. Tolerance chains of an assembly can be obtained by searching its mating graph. Xue and Ji

[18] identified tolerance chains with a surface-chain model in parameterised tolerance chart. However, the method does not involve form and position tolerances. In [19], Zhou et al. proposed a method to generate dimension chains based on a jot chain model of assembly. Hu and Wu [20] classified geometric characteristics and variational geometric constraints and set up the corresponding rules between VGCs and tolerance types. On this basis, they set up VGC theory and constructed tolerance network, which effectively solves comprehensive design of dimensional and geometric tolerances.

Establishing assembly tolerance networks in CAD systems is a complex design problem, in which there are still many unresolved issues. Some commercial CAD systems have realized the automatic generation of dimensional tolerances. However, automatic design of assembly tolerance networks cannot completely be achieved. Current researches on tolerance networks do not meet the integration requirements of CAD/CAM systems. The aim of this paper is to propose a new mathematical method for establishing assembly tolerance networks based on PS theory [21] in computer systems. PS theory can use a unified mathematic expression to describe the associations between different scales, which provides a powerful system tool for resolving multiscale issues. By way of human-computer interaction, assembly information of a mechanical assembly is extracted from its three-dimensional model in CAD systems. The nonlinear assembly sequences of the assembly are generated according to related set rules. Based on this, a hierarchical assembly structure tree model of the mechanical assembly is established. According to VGC theory [20] and PS theory [21], four relation models and the systems of location relation equations and interference relation equations are established, by using which the generation of assembly tolerance types is realized. Analyzing the construction of assembly tolerance network, assembly tolerance chains between parent nodes and child nodes are set up in the assembly structure tree. Finally, the assembly tolerance networks of mechanical assemblies are constructed in the foundation of the hierarchical assembly structure tree model. On the premise of automatically extracting assembly information, the generation of assembly tolerance types is confined to subassemblies, which effectively reduces the scale of the problem and the complexity of computing and lays a good foundation for the automatic generation of assembly tolerance types of mechanical assemblies. Meanwhile, the establishment of assembly tolerance network, which is constructed based on VGC theory and PS theory, makes a useful exploration for the qualitative design of assembly tolerance types of mechanical assemblies in CAD systems.

The paper is organized as follows. Section 2 establishes an assembly structure tree model and four relation models based on PS theory and meanwhile sets up the systems of location relation equations and interference relation equations. Based on TTRS theory and contour matrices of PS theory, Section 3 builds the reasoning matrices of VGCs and corresponding assembly tolerance types. In Section 4, an exemplary mechanical assembly is used to demonstrate the constructing process of assembly tolerance networks and simultaneously to verify the feasibility and effectiveness of the proposed method.

2. Relation Models and Systems of Relation Equations

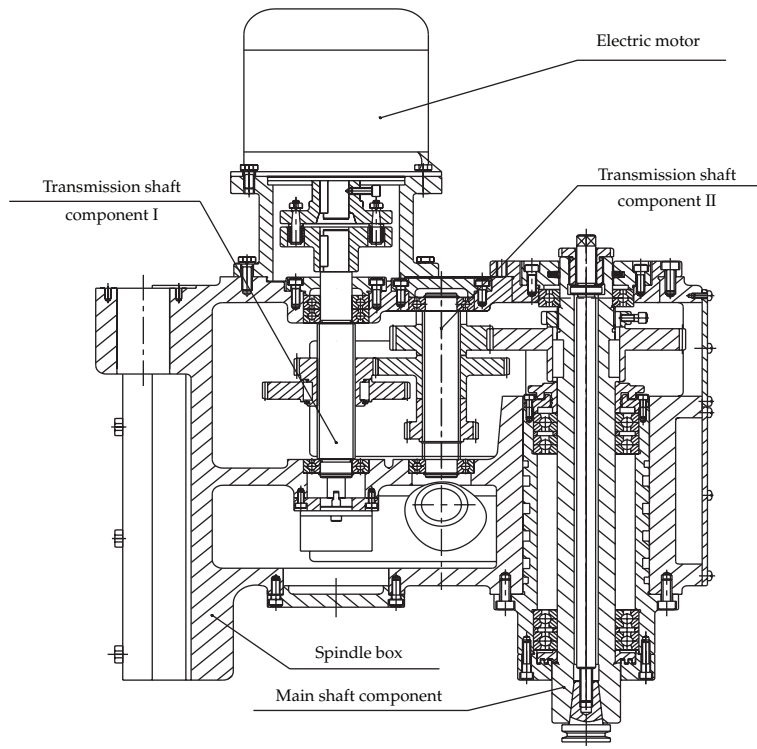
2.1. Assembly Structure Tree Model

For mechanical assemblies, their assembly processes are just like constructing objects with building blocks. Firstly, parts are assembled into components according to constraint relations

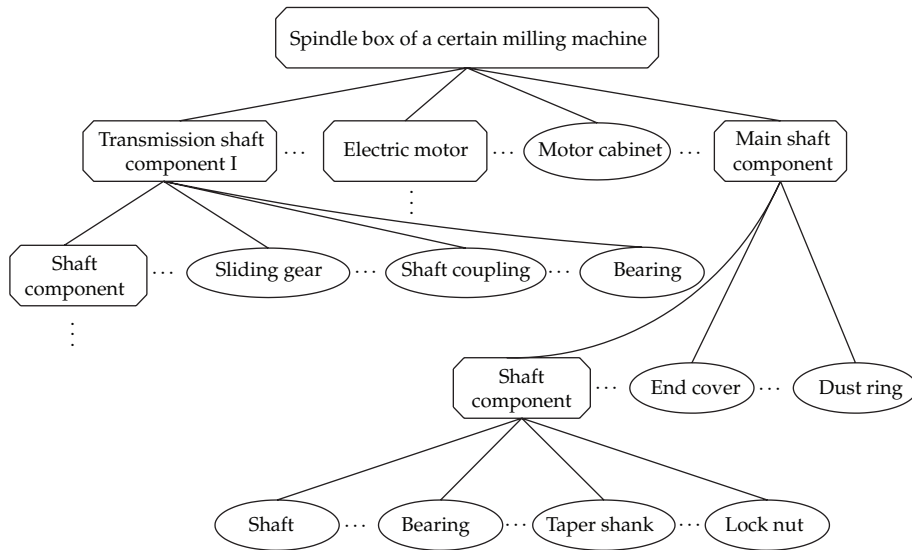
among them, and then these components and other parts are further assembled into high-level components. The above process is repeatedly carried out until the assembling of product is completed. For the majority of assemblies, their structures could be seen as follows: an assembly is composed of subassemblies and parts. Likewise, a subassembly is also composed of its subassemblies and parts directly under it. Therefore, an assembly could be disassembled into basic structural units, namely, parts. Different subassemblies and parts can be concurrently assembled. Obviously, the assembly process is a typical nonlinear process. From the above analysis, we can see that some mechanical assemblies have hierarchical structures; therefore, their structures can be expressed by product structure trees (PST). For example, the spindle box of a NC milling machine shown in Figure 1(a) just has such a structure. Its parts and subassemblies can be expressed in the different layers of a structure tree model, as shown in Figure 1(b). The assembly relations among the parts and subassemblies can also be clearly represented in the model. Hierarchical tree structure model not only expresses the structure of product but also implies its partial assembly information.

In this paper, a hierarchical structure tree model, namely, assembly structure tree (AST), is established to express assembly structure and assembly sequence of product. PST possesses a tree structure, which consists of all the parts of a product and reflects its functional relations and assembly relations. Based on the characteristics of PST and basic requirements of product assembly, we establish the following rules to reconstruct PST to obtain AST of product.

- (i) Rule 1: connection relations between parts can be divided into two types: contact and fitting. Contact includes physical contact and virtual contact. Fitting includes plane fitting, column fitting, conical fitting, spherical surface fitting, prism fitting, screw thread fitting, welding fitting, riveting fitting, and bonding fitting. Among them, connections between parts, formed by welding, riveting, and bonding, are not allowed to be dismantled; otherwise, it will result in the failure of connection form. Assembly tolerance mainly reflects assembly position precision between parts. Parts, which are connected together by bonding, riveting, or welding, should be considered as an independent part to be studied. This paper mainly considers plane fitting, column fitting, conical surface fitting, spherical surface fitting, prism fitting, and screw thread fitting.
- (ii) Rule 2: in AST, subassemblies or parts, which belong to a same parent node, cannot interfere in each other's assembling. If the interferences occur when reconstructing product structure tree, the interference components should further be broken into smaller ones at the same level, or all the subassemblies and parts in the same parent node are redivided and recombined until the assembling interferences disappear.
- (iii) Rule 3: assembling positions of parts can be determined by other parts or components' constraints in the same subassembly, and they can be assembled without interferences. Meantime, subassemblies, as a whole, can also be positioned and assembled through the contacting and fitting between them and other subassemblies or parts. They belong to the same parent node.
- (iv) Rule 4: in subassemblies, assembling positions of some parts are determined by other parts or components which do not belong to the subassemblies. These parts not only can determine the assembling positions of the subassemblies in the product but also are the assembling datums of the subassemblies. They are referred to as basic parts. In most cases, assembling process of subassembly should begin with basic parts. Under certain conditions, they can be concurrently assembled.



(a) Cutaway view of spindle box



(b) Product structure tree model

Figure 1: Spindle box of a NC milling machine.

Assembly sequence planning of subassemblies can begin with basic parts selected in the way of man-machine interaction.

- (v) Rule 5: in AST, an assembly subsequence exists among subassemblies belonging to the same parent node. Each subassembly as a whole is assembled with other subassemblies, and they all can independently realize functional requirements of product. The assembling of subassemblies in the same parent node has precedence relations.

2.2. Assembly Information Model

The assembly information model established in this paper includes geometry information and fitting information of parts. According to function, structure, and geometric shape of part, geometry information can be divided into four types: axle sleeve, wheel disc, fork, and shell. Fitting information consists of two parts: fitting type and fitting property. Based on Rule 1 mentioned above, fitting type includes plane fitting, column fitting, conical surface fitting, spherical surface fitting, prism fitting, and screw thread fitting. Fitting property can be classified into clearance fitting, transition fitting, and interference fitting.

Firstly, on the premise of automatically extracting assembly information, assembly information of parts is extracted from the three-dimensional model of product in CAD system. And then a mathematical model based on PS is established to describe the assembly information, in which the parts of assembly are used as the elements of PS and their geometric information and fitting information are used as the contour of PS. Finally, the assembly information can be formally expressed as follows:

$$\|c_{i(j)}\|_{A,F(A)} = [A \times F(A)] = \begin{bmatrix} F_1 & F_2 & \cdots & F_m \\ c_{1(1)} & c_{1(2)} & \cdots & c_{1(m)} \\ c_{2(1)} & c_{2(2)} & \cdots & c_{2(m)} \\ \vdots & \vdots & & \vdots \\ c_{n(1)} & c_{n(2)} & \cdots & c_{n(m)} \end{bmatrix} \begin{matrix} a_1 \\ a_2 \\ \vdots \\ a_n \end{matrix}, \quad (2.1)$$

where the element set of PS is

$$A = \{a_i \mid i = 1, 2, \dots, n\}, \quad (2.2)$$

a_i represents parts or components of assembly. The contour of PS is the set

$$F(A) = \{F_j \mid j = 1, 2, \dots, m\}. \quad (2.3)$$

Its element F_j represents geometric and fitting features. In the matrix, if element a_i has corresponding relation with F_j , then $c_{i(j)} = 1$.

Based on the above analysis, a formal model of assembly information is built up as follows.

In Figure 2, F_1 – F_4 represent part types, which are, respectively, axle sleeve, wheel disc, fork, and shell. F_5 – F_{10} , respectively, represent plane fitting, column fitting, conical surface

	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}
a_1	•				•	•					•		
a_2	•				•							•	
\vdots													
a_i			•		•	•				•	•	•	•
\vdots													
a_{n-1}		•				•							•
a_n	•						•				•		

Figure 2: Assembly information model.

fitting, spherical surface fitting, prism fitting, and screw thread fitting. F_{11} – F_{13} represent fitting properties of parts, namely, clearance fit, transition fit, and interference fit. The solid circle • represents that element a_i , ($i = 1, 2, \dots, n$) possesses feature F_j ($j = 1, 2, \dots, 13$).

The establishment of an assembly information model could provide basic information for subsequently setting up other models and equations.

2.3. Assembly Relation Model

If without being constrained by other parts, each part has six degrees of freedom (DOF) in free space, namely, three translational DOFs and three rotational DOFs, which translate and rotate, respectively, along the three mutually perpendicular coordinate axes. Except for the DOFs which are used to realize product functions, if all the other DOFs of a part are limited in assembly space, then its position is also determined. This implies that location requirement of the part is satisfied, which is realized by means of other related parts and components in the same subassembly. These parts and components have contact and location relations with the part and can limit its DOFs. When calculating feasible assembly sequences of a subassembly, we only need to consider assembly relations between parts which constitute the subassembly. Therefore, contact and location relations between parts in the directions of six DOFs are needed to be described in the assembly relation model.

In addition, whether a part can be assembled is also affected by other factors. For example, the space occupied by other parts probably interferes with the assembly path of the part, which makes it impossible to move the part to assembling position, interferences between a part and its assembling tools might occur if operational space is not large enough to use the assembly tools, assembling fixture interferes with assembling of parts because of its clamping method and the space occupies by it, and considering factors of man-machine engineering, there are interferences between man, part, assembly fixture, and machine. To simplify the analysis, this paper mainly considers interferences which exist between parts (or components). Consequently, assembly interference relations between parts are also needed to be described in the assembly relation model.

In an assembly, there are connection relations between parts. If not considering concrete forms of connection structures, location relations between parts can also reflect their connection relations. Thus, this paper no longer discusses connection relations of parts in detail.

By making use of hierarchical structure of the assembly structure tree, assembly information models of subassemblies in different layers could be set up. For subassembly, $S_A = \{a_i \mid i = 1, 2, \dots, n\}$ (a_i represents a part or a component belonging to the subassembly

and n is the total number of parts of the subassembly), its assembly process begins with two parts which contact each other. The two parts can be regarded as an assembly unit, and then other parts are added into it in turn under constraint conditions. Finally, the parts are assembled into a subassembly. With the same method, all the subassemblies in different levels of the assembly structure tree can be formed. This process is continuously carried out until a product is completely assembled. Based on PS theory, this paper extracts assembly information, respectively, from the assembly information model constructed above and three-dimensional model of product in CAD system to establish the assembly relation model of subassembly shown in Figure 3, in which the combination of any two parts is taken as the element of PS and the DOFs of part in assembly space are taken as the contour of PS.

Binary array (a_i, a_j) is the combination of any two parts or components in subassembly, which represents constraint relation imposed on element a_i by element a_j ; $F_{14}^L-F_{19}^L$ represent location relations between two parts, which are three translational DOFs and three rotational DOFs, respectively, along X-axis, Y-axis, and Z-axis. $F_{20}^I-F_{25}^I$ represent interference relations between two parts, respectively, along the positive and negative directions of X-axis, Y-axis, and Z-axis. The solid circle \bullet represents that there are assembly constraint relations between two parts.

2.4. Location Relation Model and System of Location Relation Equations

Positioning information of a part, which is extracted from an assembly relation model, can be used to establish its location relation model shown as Figure 4, in which $a_i, a_j \in S_A$ ($i, j \in \{1, 2, \dots, n\}$, $i \neq j$). In the location relation model, the location relation between Parts a_i and a_j is represented as function value of $\bigwedge_{k=14}^{19} F_k^L(a_i, a_j)$ where $1 \leq i, j \leq n$, $i \neq j$. If the assembly position of Part a_i is determined by Part a_j , then

$$\bigwedge_{k=14}^{19} F_k^L(a_i, a_j) = 1, \quad (2.4)$$

otherwise

$$\bigwedge_{k=14}^{19} F_k^L(a_i, a_j) = 0. \quad (2.5)$$

If the assembly position of Part a_i is determined by several related parts, then

$$\bigwedge_{k=14}^{19} \left[F_k^L(a_i, a_j) \vee F_k^L(a_i, a_l) \vee \dots \vee F_k^L(a_i, a_p) \right] = 1, \quad (2.6)$$

where $1 \leq i, j, l, p \leq n$, $i \neq j \neq l \neq p$, and n is the total number of parts in the subassembly. In the location relation model, we use the symbol \bullet to represent the value 1.

In the subassembly, if there is a group of parts which constrains 6 DOFs of a part, then a logic equation composed by this group of parts can be regarded as a location relation

	F_{14}^L	F_{15}^L	F_{16}^L	F_{17}^L	F_{18}^L	F_{19}^L	F_{20}^I	F_{21}^I	F_{22}^I	F_{23}^I	F_{24}^I	F_{25}^I
(a_1, a_2)	●				●	●	●	●				●
(a_1, a_3)	●				●							●
⋮												
(a_1, a_n)			●		●	●					●	●
(a_2, a_3)		●			●	●						●
(a_2, a_4)		●		●					●	●		
⋮												
(a_i, a_j)			●	●	●							●
(a_i, a_{j+1})		●		●	●				●	●		
⋮												
(a_{n-2}, a_n)		●		●	●	●	●		●	●		
(a_{n-1}, a_n)	●						●				●	

Figure 3: Assembly relation model.

	F_{14}^L	F_{15}^L	F_{16}^L	F_{17}^L	F_{18}^L	F_{19}^L
(a_i, a_{i+1})		●			●	●
(a_i, a_{i+2})		●		●		
⋮						
(a_i, a_j)			●	●	●	
(a_i, a_{j+1})		●		●	●	
⋮						
(a_i, a_{n-1})		●				●
(a_i, a_n)	●					

Figure 4: Location relation model of parts.

equation of this part and its logical value is used to judge whether the position of the part is determined. The formal expression for location relation is shown as follows:

$$B^L(a_i) = a_j \wedge (\vee) a_k \wedge (\vee) \cdots (a_l \wedge (\vee) a_m) \cdots, \tag{2.7}$$

where the parameters, such as a_i, a_j are the parts of the subassembly, and the AND/OR operators reflect location relations between them. Its algorithm rules are defined as follows.

(i) Rule 1: it is assumed that a_i, a_j are two parts belonging to the subassembly. If

$$\bigwedge_{k=14}^{19} F_k^L(a_i, a_j) = 1, \quad (2.8)$$

then

$$B^L(a_i) = a_j. \quad (2.9)$$

(ii) Rule 2: if

$$\bigwedge_{k=14}^{19} \left[F_k^L(a_i, a_j) \vee F_k^L(a_i, a_m) \right] = 1, \quad (2.10)$$

then

$$B^L(a_i) = a_j \wedge a_m. \quad (2.11)$$

(iii) Rule 3: if

$$\bigwedge_{k=14}^{19} F_k^L(a_i, a_j) = 1 \quad (2.12)$$

and meantime

$$\bigwedge_{k=14}^{19} F_k^L(a_i, a_m) = 1, \quad (2.13)$$

then

$$B^L(a_i) = a_j \vee a_m. \quad (2.14)$$

(iv) Rule 4: if

$$B^L(a_i) = a_j \wedge a_l, \quad (2.15)$$

$$B^L(a_i) = a_j \wedge a_m,$$

then

$$B^L(a_i) = a_j \wedge (a_l \vee a_m). \quad (2.16)$$

	F_{20}^I	F_{21}^I	F_{22}^I	F_{23}^I	F_{24}^I	F_{25}^I
(a_i, a_{i+1})						●
(a_i, a_{i+2})			●	●		
⋮						
(a_i, a_j)					●	●
(a_i, a_{j+1})			●	●		
⋮						
(a_i, a_{n-1})			●		●	●
(a_i, a_n)	●				●	

Figure 5: Interference relation model of parts.

- (v) Rule 5: it is prescribed that the position constraints, which are added to Part a_i by Part a_j , are equivalent to those which are added to Part a_j by Part a_i , that is to say

$$\bigwedge_{k=14}^{19} F_k^I(a_i, a_j) = \bigwedge_{k=14}^{19} F_k^I(a_j, a_i). \quad (2.17)$$

In order to simplify the structure of the location relation model, it is stipulated that element (a_i, a_j) must satisfy the condition $i < j$.

- (vi) Rule 6: location relation equations of all the parts in a subassembly are combined together to constitute the system of location relation equations of the subassembly.

2.5. Interference Relation Model and System of Interference Relation Equations

By means of assembly interference information of a part which is extracted from the assembly relation model, its interference relation model is set up shown in Figure 5, in which $a_i, a_j \in S_A$ ($i, j \in \{1, 2, \dots, n\}, i \neq j$), and n represents the total number of parts of subassembly. For the array element (a_i, a_j) , $F_{20}^I - F_{25}^I$ represent interference relations, respectively, along the directions of $\pm X$, $\pm Y$, and $\pm Z$ axes, which are imposed on Part a_i by Part a_j when assembling Part a_i .

In the interference relation model, the function $\bigwedge_{k=20}^{25} F_k^I(a_i, a_j)$ is used to express assembly interference relations between Part a_i and Part a_j . The algorithm rules of interference relation are established as follows.

- (i) Rule 1: when assembling Part a_i into the subassembly, if Part a_j hinders the assembly path of Part a_i , then

$$\bigwedge_{k=20}^{25} F_k^I(a_i, a_j) = 1 \quad (2.18)$$

and the interference relation equation of Part a_i is $B^I(a_i) = a_j$, otherwise

$$\bigwedge_{k=20}^{25} F_k^I(a_i, a_j) = 0, \quad (2.19)$$

$$B^I(a_i) = 0.$$

(ii) Rule 2: it is presumed that $a_i, a_j, a_l \in S_A, i, j, l \in \{1, 2, \dots, n\}$, and $i \neq j \neq l$. If Part a_j and Part a_l jointly hinder the assembly path of Part a_i , then

$$\bigwedge_{k=20}^{25} [F_k^I(a_i, a_j) \vee F_k^I(a_i, a_l)] = 1, \quad (2.20)$$

$$B^I(a_i) = a_j \wedge a_l.$$

(iii) Rule 3: If Part a_j and Part a_l , respectively, interfere with the assembly path of Part a_i , that is to say

$$\bigwedge_{k=20}^{25} F_k^I(a_i, a_j) = 1, \quad (2.21)$$

$$\bigwedge_{k=20}^{25} F_k^I(a_i, a_l) = 1,$$

then

$$B^I(a_i) = a_j \vee a_l. \quad (2.22)$$

(iv) Rule 4: if

$$B^I(a_i) = a_j \wedge a_l, \quad (2.23)$$

$$B^I(a_i) = a_j \wedge a_l,$$

then

$$B^I(a_i) = a_j \wedge (a_l \vee a_m). \quad (2.24)$$

- (v) Rule 5: owing to the direction of interference relation in assembly space, the interference constraints imposed on Part a_i by Part a_j do not mean those which are imposed on Part a_j by Part a_i . Therefore,

$$\bigwedge_{k=20}^{25} F_k^I(a_i, a_j) \neq \bigwedge_{k=20}^{25} F_k^I(a_j, a_i). \quad (2.25)$$

It is obvious that the following expressions are equivalent

$$\begin{aligned} F_{20}^I(a_i, a_j) &= F_{21}^I(a_j, a_i), \\ F_{22}^I(a_i, a_j) &= F_{23}^I(a_j, a_i), \\ F_{24}^I(a_i, a_j) &= F_{25}^I(a_j, a_i). \end{aligned} \quad (2.26)$$

In other words, the interference relations between two parts, which are, respectively, along the positive and negative directions of the same coordinate axis, are equivalent.

- (vi) Rule 6: interference relation equations of all the parts in a subassembly are combined together to constitute the system of interference relation equations of this subassembly.

3. Reasoning Methods of VGCs and Corresponding Tolerance Types

TTRS theory proposed by Salomons [22] divides functional surfaces of parts into seven basic types, which are, respectively, spherical surface, cylindrical surface, plane, helicoidal surface, rotating surface, prismatic surface, and complex surface. On this basis, Hu and Wu [20] proposed VGC theory. The theory considers that geometric constraints are constraints between nominal features. From the viewpoint of manufacturing, geometric constraints between features are variable; therefore, they are called VGC. A VGC consists of three parts: referenced feature (RF), constrained feature (CF), and variational geometric constraint (VGC). VGCs represent constraint relations between RFs and CFs. According to the differences between RF and CF, VGC can be divided into three types: self-referenced VGC (SVGC), cross-referenced VGC (CVGC), and mating VGC (MVGC). There are reasoning relations between the three kinds of VGCs and assembly tolerance types. Based on VGC theory [20] and PS theory [21, 23], this paper constructs the following matrices to describe the reasoning relations between VGCs and corresponding assembly tolerance types.

3.1. Reasoning Matrices of SVGCs and Corresponding Tolerance Types

Each SVGC is a constraint between a real feature and its corresponding associated derived feature (ADF). Functional surfaces of parts can be divided into seven types. Therefore, constraints between functional surfaces and corresponding ADFs can also be classified into

	S	Cy	Pl	H	R	Pr	Co
SC1	●						
SC2		●					
SC3			●				
SC4				●			
SC5					●		
SC6						●	
SC7							●

Figure 6: Reasoning matrix of SVGCs.

	SC1	SC2	SC3	SC4	SC5	SC6	SC7
AT1		●	●		●	●	
AT2			●				
AT3	●	●			●		
AT4		●					
AT5					●	●	●
AT6					●	●	●
AT7	●	●					

Figure 7: Reasoning matrix of tolerance types corresponding to SVGCs.

seven types. By means of the contour matrix of PS, the reasoning matrix of SVGCs is established, as shown in Figure 6. SVGCs are used as the elements of PS, and functional surfaces are used as the contour of PS. S represents spherical surface; Cy represents cylindrical surface; Pl represents plane; H represents helicoidal surface; R represents rotating surface; Pr represents prismatic surface; Co represents complex surface. SC1–SC7 represent seven types of SVGCs corresponding to the related functional surfaces. The solid circle ● represents that the corresponding relation exists between a functional surface and an SVGC.

According to the definition of SVGCs, it is obvious that there are corresponding relations between SVGCs, form tolerances, and dimensional tolerances. By using contour matrix of PS, the reasoning matrix of tolerance types corresponding to SVGCs is set up as shown in Figure 7. Tolerance types are taken as the elements of PS, in which AT1 represents straightness, AT2 represents flatness, AT3 represents circularity, AT4 represents cylindricity, AT5 represents the profile of a line, AT6 represents the profile of a surface, and AT7 represents dimensional tolerance.

3.2. Reasoning Matrices of CVGCs and Corresponding Tolerance Types

Each CVGC is a constraint between two ADFs, in which the two ADFs belong to the same part. Geometric features of a part can be decomposed into point, line, or plane. The interrelations between point, line, and plane can be divided into 27 kinds according to spatial positions of each other. Therefore, 27 kinds of CVGCs can be generated in accordance with them. The reasoning matrix of CVGCs is built shown in Figure 8. In the reasoning matrix, Po represents point, Li represents line, and Pl represents plane. IR-SIR represents spatial relations between ADFs, which are, respectively, inclusion relation, parallel relation, vertical relation in the same plane, intersection relation in the same plane, and space intersection relation. CC1–CC6 are CVGCs taking point as referenced feature, CC7–CC17 are CVGCs taking line as referenced feature, and CC18–CC27 are CVGCs taking plane as referenced feature. The solid circle • represents that there is a corresponding relation between the constrained feature of a CVGC and the spatial position relation between two features of the CVGC.

As with SVGCs, there are corresponding relations between CVGCs and assembly tolerances. Using CVGCs as contour of PS and assembly tolerance types as elements of PS, the reasoning matrix of assembly tolerance types corresponding to CVGCs is set up, as shown in Figure 9. AT8 represents parallelism; AT9 represents verticality; AT10 represents gradient; AT11 represents coaxiality; AT12 represents symmetry; AT13 represents position accuracy; AT14 represents circular run-out; AT15 represents whole runout; AT16 represents location dimension tolerance; AT17 represents location angle tolerance. The solid circle • represents that the corresponding relation between a CVGC and an assembly tolerance type is determined.

3.3. Reasoning Matrix of MVGCs

Constraints, which exist between two real features and, meanwhile, respectively, belong to two different parts, constitute MVGCs. They are actually constraints between contact surfaces of two parts. MVGCs with lower pairs, which are often seen in assembly design, can be divided into seven types. By means of contour matrix of PS, the reasoning matrix of MVGCs is established, shown in Figure 10. Mating surfaces are used as the contour of the matrix, and their corresponding MVGCs are used as the elements of the matrix, in which S represents spherical surface, Cy represents cylindrical surface, Pl represents plane, H represents helicoidal surface, R represents rotating surface, Pr represents prismatic surface, and Co represents complex surface. MC1–MC7, respectively, represent MVGCs which correspond to related mating surfaces. The solid circle • represents that the relation between mating surface and MVGC is determined.

3.4. Mating Tree

In an assembly structure tree, each mating relation uniquely corresponds to two SVGCs and one MVGC. Four features between two assembly parts are taken as nodes, and three VGCs are taken as arc curves. The tree model is called mating tree, which can be represented with the following equation:

$$MT(S1;S2) = T(V;E), \quad (3.1)$$

	Po	Li	Pl	IR	PR	VR	PIR	SIR
CC1	•			•				
CC2	•							•
CC3		•		•				
CC4		•						•
CC5			•	•				
CC6			•					•
CC7	•			•				
CC8	•							•
CC9		•		•				
CC10		•			•			
CC11		•				•		
CC12		•					•	
CC13		•						•
CC14			•	•				
CC15			•		•			
CC16			•			•		
CC17			•				•	
CC18	•			•				
CC19	•							•
CC20		•		•				
CC21		•			•			
CC22		•				•		
CC23		•					•	
CC24			•	•				
CC25			•		•			
CC26			•			•		
CC27			•				•	

Figure 8: Reasoning matrix of CVGCs.

	CC1	CC2	CC3	CC4	CC5	CC6	CC7	CC8	CC9	CC10	CC11	CC12	CC13	CC14	CC15	CC16	CC17	CC18	CC19	CC20	CC21	CC22	CC23	CC24	CC25	CC26	CC27
AT8										•						•						•				•	
AT9											•						•						•			•	
AT10												•						•						•			•
AT11	•									•																	
AT12														•							•				•		
AT13		•	•	•	•	•	•	•		•			•	•	•												
AT14									•																		
AT15									•																		
AT16		•		•		•		•		•			•	•	•	•	•	•									
AT17												•						•						•			•

Figure 9: Reasoning matrix of tolerance types corresponding to CVGCs.

	S	Cy	Pl	H	R	Pr	Co
MC1	•						
MC2		•					
MC3			•				
MC4				•			
MC5					•		
MC6						•	
MC7							•

Figure 10: Reasoning matrix of MVGCs.

where MT represents the mating tree. S1 and S2 are two features mating each other. V is the set of two real features and two ADFs. The two real features mate each other and the two ADFs, respectively, correspond to the two real features. E is the set of SVGCs and MVGCs. The two ADFs can separately constrain their corresponding RFs by SVGCs, and the two RFs can constrain each other by MVGCs.

4. Assembly Tolerance Network

By using the reasoning methods described above, we can reason out assembly sequence of all the subassemblies in different layers of assembly structure tree and tolerance types between parts or subassemblies and then add them into the assembly structure tree. This kind of structure tree with assembly sequence and assembly tolerance information is defined as assembly tolerance network. In order to simplify the process of constructing the tolerance network, we take a subassembly of the spindle box of an NC milling machine (shown in Figure 1), namely, the main shaft component, as an example to describe the constructing process. Constructing processes related to other subassemblies and parts are similar to it. The concrete structure of the main shaft component is represented in Figure 11. Assembly sequence of a fastener can be determined by assembly sequence of a part or parts group connected by it. Therefore, this paper does not discuss the assembling of fasteners. The fasteners in Figure 11 are not marked.

- (i) Step 1: extract assembly information. On the premise of automatically recognizing features, the basic information of the main shaft component is extracted from its three-dimensional model in CAD system.
- (ii) Step 2: establish assembly structure tree. According to the rules in Section 2.1, the assembly structure tree of the main shaft component is built up, shown in Figure 12. It has a three-layer structure. The bearing groups consist of two bearings and one space collar (see Figure 11), whose assembly sequences are fixed. Therefore, this paper regards Bearing group 1 and Bearing group 2 as parts to simplify the analysis process. The dust ring is not considered here because it is a flexible part and do not have direct relations with assembly tolerances.
- (iii) Step 3: set up assembly information model. According to the rules in Section 2.2, the assembly information model of the main shaft component is established, shown in Figure 13. P_1 represents the bearing; P_2 represents the end cover 3; P_3 represents the end cover 2; P_4 represents the locknut; P_5 represents the gear; P_6 represents the shaft key; P_7 represents the bearing group 1; P_8 represents the end cover 1; P_9 represents the baffle plate; P_{10} represents the shaft sleeve; P_{11} represents the bearing group 2; P_{12} represents the dust cap 1; P_{13} represents the dust cap 2; P_{14} represents the main shaft; P_{15} represents the taper shank; P_{16} represents the nut; P_{17} represents the pull rod.
- (iv) Step 4: establish assembly relation model. On the basis of Section 2.3, we establish the assembly relation models of the main shaft component and shaft component, shown in Figure 14. The shaft component is represented with C_1 .
- (v) Step 5: establish the system of location relation equations. Extract location information from the assembly relation models in Figure 14, and then set up the system of location relation equations according to the rules in Section 2.4.

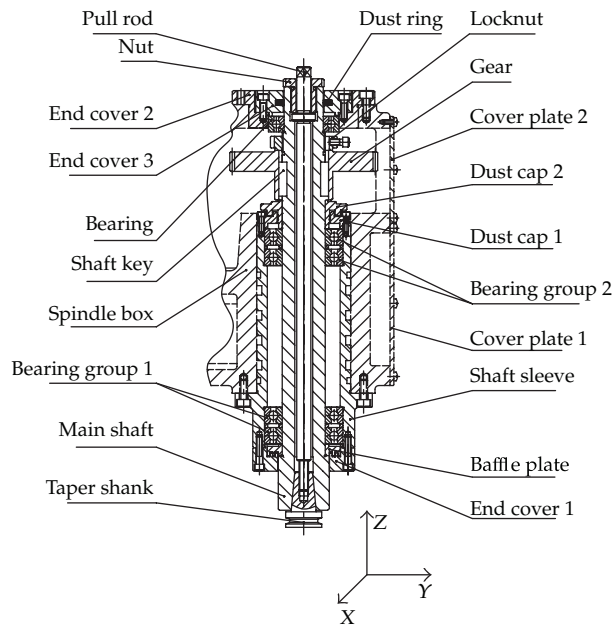


Figure 11: Main shaft component.

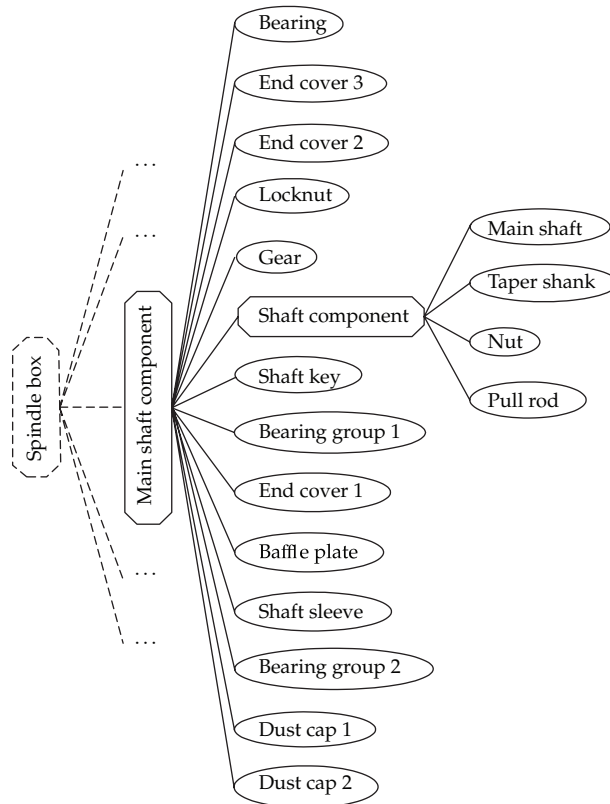


Figure 12: Assembly structure tree of main shaft component.

	F_1	F_2	F_3	F_4	F_5	F_6	F_7	F_8	F_9	F_{10}	F_{11}	F_{12}	F_{13}
P_1	•				•	•							•
P_2	•				•						•		
P_3	•				•	•					•		
P_4	•				•					•			
P_5		•			•	•					•		
P_6	•				•						•		
P_7	•				•	•							•
P_8	•				•	•					•		
P_9	•				•	•					•		
P_{10}	•				•	•					•		
P_{11}	•				•	•							•
P_{12}	•				•	•					•		
P_{13}	•				•	•					•		
P_{14}	•				•	•					•		•
P_{15}	•						•			•		•	
P_{16}	•	•								•			
P_{17}	•				•	•					•		

Figure 13: Assembly information model of main shaft component.

The system of location relation equations of the main shaft component is listed as follows:

$$\begin{aligned}
 B^L(P_1) &= C_1; & B^L(P_2) &= P_3 \wedge C_1; & B^L(P_3) &= P_1; & B^L(P_4) &= P_5 \wedge C_1, \\
 B^L(P_5) &= P_6 \wedge P_{13} \wedge C_1; & B^L(P_6) &= C_1; & B^L(P_7) &= (P_9 \vee P_{10}) \wedge C_1, \\
 B^L(P_8) &= P_{10}; & B^L(P_9) &= C_1; & B^L(P_{10}) &= P_7 \wedge P_{11}, \\
 B^L(P_{11}) &= P_{10} \wedge C_1; & B^L(P_{12}) &= P_{10} \wedge P_{11}, \\
 B^L(P_{13}) &= P_{11} \wedge C_1; & B^L(C_1) &= C_1.
 \end{aligned} \tag{4.1}$$

The system of location relation equations of the shaft component is listed as follows:

$$\begin{aligned}
 B^L(P_{14}) &= P_{14}; & B^L(P_{15}) &= P_{14}, \\
 B^L(P_{16}) &= P_{14}; & B^L(P_{17}) &= P_{14} \wedge P_{16}
 \end{aligned} \tag{4.2}$$

	F_{14}^L	F_{15}^L	F_{16}^L	F_{17}^L	F_{18}^L	F_{19}^L	F_{20}^L	F_{21}^L	F_{22}^L	F_{23}^L	F_{24}^L	F_{25}^L
(P_1, P_2)			•	•	•				•			
(P_1, P_3)	•	•		•	•		•	•		•	•	
(P_1, C_1)	•	•		•	•		•	•		•	•	•
(P_2, P_3)			•	•	•		•	•		•	•	•
(P_2, C_1)	•	•		•	•		•	•		•	•	
(P_4, P_5)			•	•	•				•			
(P_4, C_1)	•	•	•	•	•		•	•		•	•	
(P_5, P_{13})			•	•	•							•
(P_5, C_1)	•	•		•	•		•	•		•	•	
(P_6, C_1)	•		•	•	•	•	•		•	•		•
(P_7, P_9)			•	•	•				•			•
(P_7, P_{10})	•	•	•	•	•		•	•	•	•	•	
(P_7, C_1)	•	•	•	•	•		•	•	•	•	•	
(P_8, P_{10})	•	•	•	•	•		•	•		•	•	•
(P_9, C_1)	•	•	•	•	•		•	•	•	•	•	
(P_{10}, P_{11})	•	•	•	•	•		•	•		•	•	•
(P_{10}, P_{12})	•	•	•	•	•		•	•		•	•	•
(P_{11}, P_{13})			•	•	•							•
(P_{11}, C_1)	•	•	•	•			•	•		•	•	
	F_{14}^L	F_{15}^L	F_{16}^L	F_{17}^L	F_{18}^L	F_{19}^L	F_{20}^L	F_{21}^L	F_{22}^L	F_{23}^L	F_{24}^L	F_{25}^L
(P_{14}, P_{15})	•	•		•	•		•	•		•	•	•
(P_{14}, P_{16})	•	•	•	•	•	•	•	•	•	•	•	•
(P_{14}, P_{17})	•	•		•	•		•	•		•	•	•
(P_{16}, C_{17})			•	•	•		•	•		•	•	•

Figure 14: Assembly related models of main shaft component and shaft component. (a) Assembly relation model of main shaft component. (b) Assembly relation model of shaft component.

- (vi) Step 6: establish the system of interference relation equations. Like Step 5, extract the location information from the assembly relation models in Figure 14, and then set up the system of interference equations according to the rules in Section 2.5.

The system of interference relation equations of the main shaft component is listed as follows:

$$B^I(P_1) = P_2; \quad B^I(P_2) = 0; \quad B^I(P_3) = P_2,$$

$$B^I(P_4) = (P_1 \vee P_2 \vee P_3) \wedge C_1,$$

$$\begin{aligned}
B^I(P_5) &= (P_1 \vee P_2 \vee P_3 \vee P_4) \wedge C_1, \\
B^I(P_6) &= (P_1 \vee P_2 \vee P_3 \vee P_4) \wedge P_5 \wedge C_1, \\
B^I(P_7) &= (P_9 \vee C_1) \wedge P_{10}; \quad B^I(P_8) = 0, \\
B^I(P_9) &= (P_1 \vee P_2 \vee P_3 \vee P_4 \vee P_5 \vee P_6 \vee P_{10} \vee P_{11} \vee P_{12} \vee P_{13}) \wedge C_1, \\
B^I(P_{10}) &= (P_8 \vee P_9) \wedge C_1, \\
B^I(P_{11}) &= (P_1 \vee P_2 \vee P_3 \vee P_4 \vee P_5 \vee P_6 \vee P_{12} \vee P_{13}) \wedge C_1, \\
B^I(P_{12}) &= (P_1 \vee P_2 \vee P_3 \vee P_4 \vee P_5 \vee P_6 \vee P_{13}) \wedge C_1, \\
B^I(P_{13}) &= (P_1 \vee P_2 \vee P_3 \vee P_4 \vee P_5) \wedge C_1, \\
B^I(C_1) &= (P_1 \vee P_2 \vee P_3 \vee P_4 \vee P_5 \vee P_6 \vee P_7 \vee P_9 \vee P_{10} \vee P_{12} \vee P_{13}) \wedge (P_{10} \vee P_{11} \vee P_{13}).
\end{aligned} \tag{4.3}$$

The system of interference relation equations of the shaft component is listed as follows:

$$\begin{aligned}
B^I(P_{14}) &= P_{15} \wedge (P_{16} \vee P_{17}), \\
B^I(P_{15}) &= 0, \\
B^I(P_{16}) &= 0, \\
B^I(P_{17}) &= P_{14} \wedge P_{16}.
\end{aligned} \tag{4.4}$$

(vii) Step 7: generate assembly sequences. Using the reasoning equations generated above and related reasoning method [24, 25], we can obtain ten assembly sequences of the main shaft component and three assembly sequences of the shaft component. The three assembly sequences of the main shaft component are shown as follows:

$$\begin{aligned}
&C_1 \longrightarrow P_9 \longrightarrow P_7 \longrightarrow P_{10} \longrightarrow P_{11} \longrightarrow P_{12} \longrightarrow P_{13} \longrightarrow P_6 \longrightarrow P_5 \longrightarrow P_4 \\
&\quad \longrightarrow P_1 \longrightarrow P_3 \longrightarrow P_2 \longrightarrow P_8, \\
&C_1 \longrightarrow P_9 \longrightarrow P_7 \longrightarrow P_{10} \longrightarrow P_8 \longrightarrow P_{11} \longrightarrow P_{12} \longrightarrow P_{13} \longrightarrow P_6 \longrightarrow P_5 \\
&\quad \longrightarrow P_4 \longrightarrow P_1 \longrightarrow P_3 \longrightarrow P_{28}, \\
&C_1 \longrightarrow P_9 \longrightarrow P_7 \longrightarrow P_{10} \longrightarrow P_{11} \longrightarrow P_8 \longrightarrow P_{12} \longrightarrow P_{13} \longrightarrow P_6 \longrightarrow P_5 \\
&\quad \longrightarrow P_4 \longrightarrow P_1 \longrightarrow P_3 \longrightarrow P_{28}.
\end{aligned} \tag{4.5}$$

The assembly sequences of the shaft component are shown as follows:

$$\begin{aligned}
 P_{14} &\longrightarrow P_{17} \longrightarrow P_{16} \longrightarrow P_{15}, \\
 P_{14} &\longrightarrow P_{17} \longrightarrow P_{15} \longrightarrow P_{16}, \\
 P_{14} &\longrightarrow P_{15} \longrightarrow P_{17} \longrightarrow P_{16}.
 \end{aligned} \tag{4.6}$$

(viii) Step 8: determine mating tree and datum reference frame. SVGCs and MVGCs are combined into mating trees. ADFs of all the datums belonging to the same part are combined together to form a datum reference frame (DRF), between which there are no SVGCs. Taking the shaft component as an example, we use its assembly sequence, namely, $P_{14} \rightarrow P_{17} \rightarrow P_{16} \rightarrow P_{15}$, to obtain mating trees $M(P_{14}, P_{17})$, $M(P_{17}, P_{16})$, $M(P_{16}, P_{14})$, and $M(P_{14}, P_{15})$. The DRFs of P_{14} , P_{15} , P_{16} , and P_{17} are, respectively, DRF_{14} , DRF_{15} , DRF_{16} , and DRF_{17} .

(ix) Step 9: generate assembly feature chains. Taking the mating trees and datum reference frames as nodes, the subassembly sequence is reconstructed, which starts from P_{15} . The result is shown as follows:

$$DRF_{15} \longrightarrow MT(P_{15}, P_{14}) \longrightarrow DRF_{14} \longrightarrow MT(P_{14}, P_{17}) \longrightarrow DRF_{17} \longrightarrow MT(P_{17}, P_{16}) \longrightarrow DRF_{16}. \tag{4.7}$$

By means of the features of the parts in Figure 10, the above expression can be further decomposed into a feature chain as follows:

$$\begin{aligned}
 DRF_{15} &\longrightarrow P_{15_ADF_1} \longrightarrow P_{15_RF_1} \longrightarrow P_{14_RF_1} \longrightarrow P_{14_ADF_1} \\
 &\longrightarrow DRF_{14} \longrightarrow P_{14_ADF_2} \longrightarrow P_{14_RF_2} \longrightarrow P_{17_RF_1} \longrightarrow P_{17_ADF_1} \\
 &\longrightarrow DRF_{17} \longrightarrow P_{17_ADF_2} \longrightarrow P_{17_RF_2} \longrightarrow P_{16_RF_1} \longrightarrow P_{16_ADF_1} \longrightarrow DRF_{16},
 \end{aligned} \tag{4.8}$$

where ADF_1 and ADF_2 are two ADFs which belong to the same part. Likewise, RF_1 and RF_2 are two RFs which also belong to the same part.

(x) Step 10: reason VGC types between features. Using the reasoning matrices of SVGCs and CVGCs in Sections 3.2–3.4, VGC types can be reasoned out and then be added into the above expression. Finally, the expression is further decomposed into VGC chain in which the components are translations and rotations, respectively, along the directions of X , Y , and Z -axis, shown as follows:

$$\begin{aligned}
 DRF_{15} &\xrightarrow[T_x, T_y, T_z, R_x, R_y, R_z]{CC9} P_{15_ADF_1} \xrightarrow[T_x, T_y, T_z, R_x, R_y]{SC5} P_{15_RF_1} \xleftarrow[T_x, T_y, R_x, R_y]{MC5} P_{14_RF_1} \\
 &\xleftarrow[T_x, T_y, T_z, R_x, R_y]{SC5} P_{14_ADF_1} \xleftarrow[T_x, T_y, T_z, R_x, R_y, R_z]{CC9} DRF_{14} \xrightarrow[T_z, R_x, R_y]{CC25} P_{14_ADF_2}
 \end{aligned}$$

$$\begin{aligned}
& \xrightarrow[T_z, R_x, R_y]{SC3} P_{14_RF2} \xleftarrow[T_z, R_x, R_y]{MC3} P_{17_RF1} \xleftarrow[T_z, R_x, R_y]{SC3} P_{17_ADF1} \xleftarrow[T_z, R_x, R_y]{CC25} DRF17 \\
& \xrightarrow[T_z, R_x, R_y]{CC25} P_{17_ADF2} \xrightarrow[T_z, R_x, R_y]{SC3} P_{17_RF2} \xleftarrow[T_z, R_x, R_y]{MC3} P_{16_RF1} \\
& \xleftarrow[T_z, R_x, R_y]{SC3} P_{16_ADF1} \xleftarrow[T_z, R_x, R_y]{CC25} DRF16,
\end{aligned} \tag{4.9}$$

where \rightarrow represents the path from a referenced feature to a constrained feature and \leftrightarrow represents that the two features connected by it are mutually a referenced feature and a constrained feature.

- (xi) Step 11: reason tolerance types corresponding to VGCs. It can be seen that the same VGCs can correspond to different tolerance types in the reasoning matrices of tolerance types in Sections 3.2–3.4. Therefore, the related tolerance types should be chosen according to the assembly properties and functional requirements between the mating parts. By virtue of the reasoning matrices, the tolerance type corresponding with each of the VGCs can be reasoned out and then be added into the VGC chain to get the subassembly tolerance chain shown as follows:

$$\begin{aligned}
DRF15 & \xrightarrow[T_x, T_y, T_z, R_x, R_y, R_z]{CC9 \ AT11} P_{15_ADF1} \xrightarrow[T_x, T_y, T_z, R_x, R_y]{SC5 \ AT1 \ AT3} P_{15_RF1} \xleftarrow[T_x, T_y, R_x, R_y]{MC5 \ Fitting \ accuracy} P_{14_RF1} \\
& \xleftarrow[T_x, T_y, T_z, R_x, R_y]{SC5 \ AT1 \ AT3} P_{14_ADF1} \xleftarrow[T_x, T_y, T_z, R_x, R_y, R_z]{CC9 \ AT11} DRF14 \xrightarrow[T_z, R_x, R_y]{CC25 \ AT8} P_{14_ADF2} \xrightarrow[T_z, R_x, R_y]{SC3 \ AT2} P_{14_RF2} \\
& \xrightarrow[T_z, R_x, R_y]{MC3 \ Fitting \ accuracy} P_{17_RF1} \xleftarrow[T_z, R_x, R_y]{SC3} P_{17_ADF1} \xleftarrow[T_z, R_x, R_y]{CC25 \ AT8} DRF17 \xrightarrow[T_z, R_x, R_y]{CC25 \ AT8} P_{17_ADF2} \\
& \xrightarrow[T_z, R_x, R_y]{SC3 \ AT2} P_{17_RF2} \xleftarrow[T_z, R_x, R_y]{MC3 \ Fitting \ accuracy} P_{16_RF1} \xleftarrow[T_z, R_x, R_y]{SC3 \ AT2} P_{16_ADF1} \xleftarrow[T_z, R_x, R_y]{CC25 \ AT8} DRF16.
\end{aligned} \tag{4.10}$$

- (xii) Step 12: establish assembly tolerance network. In the light of the reasoning rules mentioned above, the assembly tolerance types of the spindle box and its all subassemblies can be reasoned out to form assembly tolerance chains. All the assembly tolerance chains reasoned out in Step 11 and the assembly sequence are combined to construct the assembly tolerance network of the product. Figure 15 is a schematic graph of assembly tolerance network of the spindle box, in which we can see that all the subassembly tolerance chains in dashed line frames and assembly sequences represented by the numbers with a circle are combined to form the assembly tolerance network of the whole product.

5. Conclusions

Assembly tolerance network design is determined by many factors, and these factors associate with each other. Therefore, the design is a multiscale issue. Many scholars have applied

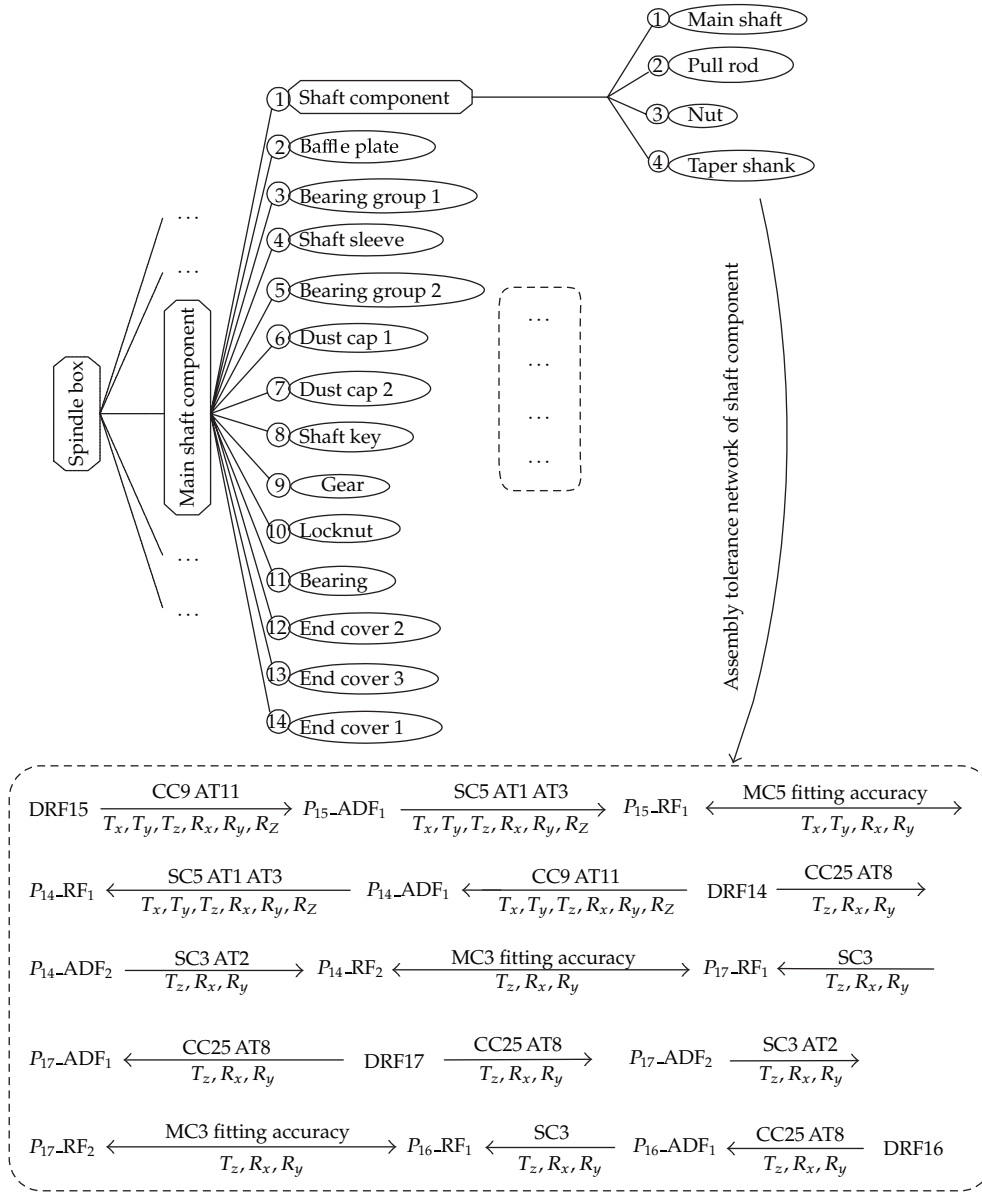


Figure 15: Assembly tolerance network of the spindle box.

various methods to resolve the multiscale issues in different research fields, such as multiscale time schemes [1], multiscale time-frequency representation [3], multiscale curvelet shrinkage [4], multiscale Retinex model [7], and multiscale nonlinear modeling algorithm [10]. Obviously, there are many factors affecting assembly tolerance design and they have great differences between each other. An effective method must be used to represent the reasoning and constraint relations. In this paper, PS theory [21] is introduced to describe the associations between different factors. Some models are established with the unified mathematic expression, including assembly information model, assembly relation model between parts, location

relation model of parts, interference relation model of parts, reasoning matrices of VGCs, and reasoning matrices of tolerance types. On this basis, this paper combines assembly tolerance design with assembly sequence planning to research the automatic generation of assembly tolerances of mechanical assemblies. By means of the product prototype in CAD system as well as the establishment of a hierarchical assembly structure tree model, this paper uses the related reasoning rules to solve the assembly sequences of the product. Then, the automatic generation of assembly tolerance types and the construction of assembly tolerance network are realized based on the variational geometric constraints theory and polychromatic sets theory. The assembly sequences, respectively, belonging to components and product are generated by the same method, which can effectively reduce the scale of the solved problem and avoid the combination explosion in the solving process. All the relation models are established by using the unified mathematic expression, which is convenient to manage knowledge better and can enhance reasoning efficiency when designing assembly tolerance networks. This method lays a good foundation for the automatic generation of assembly tolerances of mechanical assemblies. Meanwhile, the establishment of assembly tolerance network makes a useful exploration for the quantitative design of assembly tolerances of complex product.

Acknowledgments

The authors are particularly grateful to Editor Carlo Cattani for his continual encouragement, patience and sincere help on earlier drafts of this paper. The authors also would like to thank two anonymous reviewers whose insightful comments and constructive suggestions have substantially improved the paper. The authors gratefully acknowledge the support from the major project of the National Natural Science Foundation of China (NSFC) under Grant number 50935006.

References

- [1] J. Kou, S. Sun, and B. Yu, "Multiscale time-splitting strategy for multiscale multiphysics processes of two-phase flow in fractured media," *Journal of Applied Mathematics*, vol. 2011, Article ID 861905, 24 pages, 2011.
- [2] C. Picard, C. Frisson, F. Faure, G. Drettakis, and P. G. Kry, "Advances in modal analysis using a robust and multiscale method," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, Article ID 392782, 12 pages, 2010.
- [3] H. Zhu, C. Liu, and W. Gaetz, "Estimation of time-varying coherence and its application in understanding brain functional connectivity," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, Article ID 390910, 11 pages, 2010.
- [4] L. Xiao, L.-L. Huang, and B. Roysam, "Image variational denoising using gradient fidelity on curvelet shrinkage," *EURASIP Journal on Advances in Signal Processing*, vol. 2010, Article ID 398410, 16 pages, 2010.
- [5] E. G. Bakhoun and C. Toma, "Dynamical aspects of macroscopic and quantum transitions due to coherence function and time series events," *Mathematical Problems in Engineering*, vol. 2010, Article ID 428903, 13 pages, 2010.
- [6] M. Djilas, C. Azevedo-Coste, D. Guiraud, and K. Yoshida, "Spike sorting of muscle spindle afferent nerve activity recorded with thin-film intrafascicular electrodes," *Computational Intelligence and Neuroscience*, vol. 2010, Article ID 836346, 13 pages, 2010.
- [7] S. Chen and A. Beghdadi, "Natural enhancement of color image," *EURASIP Journal on Image and Video Processing*, vol. 2010, Article ID 175203, 19 pages, 2010.
- [8] C. Chen, R. Saxena, and G.-W. Wei, "A multiscale model for virus capsid dynamics," *International Journal of Biomedical Imaging*, vol. 2010, Article ID 308627, 9 pages, 2010.

- [9] L. Florack and H. Van Assen, "A new methodology for multiscale myocardial deformation and strain analysis based on tagging MRI," *International Journal of Biomedical Imaging*, vol. 2010, Article ID 341242, 8 pages, 2010.
- [10] M. N. Nounou and H. N. Nounou, "Reduced noise effect in nonlinear model estimation using multiscale representation," *Modelling and Simulation in Engineering*, vol. 2010, Article ID 217305, 8 pages, 2010.
- [11] M. N. Nounou and H. N. Nounou, "Multiscale latent variable regression," *International Journal of Chemical Engineering*, vol. 2010, Article ID 935315, 8 pages, 2010.
- [12] A. Lucia, "A multiscale gibbs-helmholtz constrained cubic equation of state," *Journal of Thermodynamics*, vol. 2010, Article ID 238365, 10 pages, 2010.
- [13] A. Clement and A. Riviere, "Tolerancing versus nominal modeling in next generation CAD/CAM system," in *Proceedings of 3rd CIRP Seminar on Computer Aided Tolerancing*, Cachan, France, April 1993.
- [14] A. Clement, A. Riviere, and P. A. Serre, "A declarative information model for functional requirements," in *Proceedings of 4th CIRP Seminars on Computer Aided Tolerancing*, Tokyo, Japan, April 1995.
- [15] P. Hoffmann, "Analysis of tolerances and process inaccuracies in discrete part manufacturing," *Computer-Aided Design*, vol. 14, no. 2, pp. 83–88, 1982.
- [16] A. Desrochers and A. Clément, "A dimensioning and tolerancing assistance model for CAD/CAM systems," *The International Journal of Advanced Manufacturing Technology*, vol. 9, no. 6, pp. 352–361, 1994.
- [17] N. Wang and T. M. Ozsoy, "Automatic generation of tolerance chains from mating relations represented in assembly models," *Journal of Mechanical Design*, vol. 115, no. 4, pp. 757–761, 1993.
- [18] J. Xue and P. Ji, "Identifying tolerance chains with a surface-chain model in tolerance charting," *Journal of Materials Processing Technology*, vol. 123, no. 1, pp. 93–99, 2002.
- [19] J. Q. Zhou, Y. F. Xing, X. M. Lai, G. L. Chen, X. Lan, and C. Z. Wei, "Study on dimension chain generation for auto-body tolerance analysis," *Journal of Shanghai Jiaotong University (Science)*, vol. 11, no. 4, pp. 417–422, 2006.
- [20] J. Hu and Z. Wu, "Methods for generation of variational geometric constraints network for assembly," *Journal of Computer-Aided Design and Computer Graphics*, vol. 14, no. 1, pp. 79–82, 2002.
- [21] Z. Li and L. D. Xu, "Polychromatic sets and its application in simulating complex objects and systems," *Computers and Operations Research*, vol. 30, no. 6, pp. 851–860, 2003.
- [22] O. W. Salomons, *Computer support in the design of mechanical products: constraint specification and satisfaction in feature based design for manufacturing*, Ph.D. thesis, University of Twente, The Netherlands, 1995.
- [23] Y. Zhang, Z. B. Li, and J. K. Wang, "Hierarchical reasoning model of tolerance information and its using in reasoning technique of geometric tolerance types," in *the International Conference of Intelligent Robotics and Applications*, Wuhan, China, October 2008.
- [24] S. Zhao and Z. Li, "Formalized reasoning method for assembly sequences based on Polychromatic Sets theory," *International Journal of Advanced Manufacturing Technology*, vol. 42, no. 9-10, pp. 993–1004, 2009.
- [25] S. Zhao and Z. Li, "A new assembly sequences generation of three dimensional product based on polychromatic sets," *Information Technology Journal*, vol. 7, no. 1, pp. 112–118, 2008.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

