Towards a tolerance representation model for generating tolerance specification schemes and corresponding tolerance zones

Yuchu Qin [1], Wenlong Lu [1,*], Qunfen Qi [2], Xiaojun Liu [1], Meifa Huang [3], Paul J Scott [2], Xiangqian Jiang [2]

[1] *The State Key Laboratory of Digital Manufacturing Equipment and Technology, School of Mechanical Science and Engineering, Huazhong University of Science and Technology, Wuhan 430074, PR China*

[2] *EPSRC Future Advanced Metrology Hub, University of Huddersfield, Huddersfield, HD1 3DH, UK*

[3] *School of Mechanical and Electrical Engineering, Guilin University of Electronic Technology, Guilin 541004, PR China*

Abstract: In this paper, a tolerance representation model for generating tolerance specification schemes and corresponding tolerance zones is proposed to meet the requirement of the representation of tolerance information semantics. This model is hierarchically organized and consists of five layers. They are component, geometric feature, variational geometric constraint, tolerance specification scheme, and tolerance zone layers. The mating relations between components in the component layer, the mating relations between geometric features in the geometric feature layer, the variational geometric constraints between geometric features in the variational geometric constraint layer, the tolerance specification schemes of component in the tolerance specification scheme layer, and the tolerance zones of tolerance specification schemes in the tolerance zone layer are formally defined by one or more adjacency matrices, respectively. Based on the model, a method for generating tolerance specification schemes for component and their resultant tolerance zones is designed. This method shows how to adopt a top-down strategy to carry out tolerance specification for an arbitrary assembly designed in a CAD system. The paper also provides a practical example to illustrate how the method works.

Keywords: Tolerance representation model; Computer-aided tolerance specification; Tolerance information semantics; Generation of tolerance specification scheme; Generation of tolerance zone; Adjacency matrix

1. Introduction

Tolerance representation model aims to reasonably and effectively represent the semantics of tolerance information in computers. It has two main requirements [1]: (1) Organizing and representing different types of tolerance information in a relatively independent way and meanwhile reflecting the difference in semantics among different types of tolerances. (2) Designing a data structure in which the geometries and dimensions required in tolerance information representation are explicitly specified and taking it as the carrier of storage and representation of tolerance information in computers. This data structure should have the capability to support computer-aided generation of tolerancing schemes.

With an aim of satisfying the two requirements, many kinds of tolerance representation models, such

* Corresponding author. *Tel.*: +86–027–87557835
E-mail address: hustwenlong@mail.hust.edu.cn (Wenlong Lu).

as mathematical definition based model, technologically and topologically related surface (TTRS) model, graph based model, modeling language model, category theory model, polychromatic set model, adjacency matrix model, and ontology-based model, have been presented during the past few decades [2], where adjacency matrix model [3] is one of them. This model consists of four layers: component layer, assembly feature layer, spatial relation layer, and tolerance type layer. The relations in each layer are formally defined with an adjacency matrix. The correspondence between assembly features and tolerance types and the correspondence between spatial relations and tolerance types are also formalized by adjacency matrices. By carrying out adjacency matrices based reasoning, tolerance types can be automatically generated. The main advantage of the model is that it is simple, intuitive, and easy to be implemented. However, the model only represents some correspondences and does not express the detailed semantics of tolerance information (i.e. the variational geometric constraints (VGCs) and resultant tolerance zones). Besides, it only focuses on the generation of tolerance types and does not consider the generation of the whole tolerancing schemes and corresponding tolerance zones.

To compensate for such deficiencies, a tolerance representation model for generating tolerance specification schemes (TSSs) and corresponding tolerance zones is proposed in this paper. Starting from the demand of representing the semantics of tolerance information, the proposed model extends the assembly feature layer, the spatial relation layer, and the tolerance type layer of the adjacency matrix model to a geometric feature layer, a VGC layer, and a TSS layer, respectively. Then it adds a tolerance zone layer after the TSS layer. By using adjacency matrices to formally define the knowledge in these layers, the model is constructed. A method for generating TSSs and corresponding tolerance zones is designed on the basis of the constructed model. Thus compared to the adjacency matrix model, the proposed model not only represents the semantics of tolerance information, but also takes into account the generation of TSSs and corresponding tolerance zones.

The remainder of the paper is organized as follows. An overview of related work is provided in Section 2. The details of the proposed model and the designed method are explained in Section 3 and Section 4, respectively. Section 5 presents a practical example to illustrate the working process of the method. In Section 6, a comparison is made and a discussion is carried out. Section 7 ends the paper with a conclusion.

2. Related work

Representation of tolerance information is the use of certain computer readable methods to represent the detailed semantics of tolerance information based on certain mathematical models of tolerance information. It generally includes the representation of VGCs that meet the tolerance requirement and the representation of resultant tolerance zones [4]. The main purpose of such representations is to construct a representation model of tolerance information semantics to enable further computer-aided tolerance specification, allocation, and analysis. Focusing on the issue of the representation of tolerance information for generating TSSs and corresponding tolerance zones, various representation models have been presented since the advent of computer-aided tolerancing (CAT) [2]. These models can be classified into mathematical definition based model, TTRS model, graph-based model, modeling language model, category theory model, polychromatic set model, adjacency matrix model, and ontology-based model according to the used knowledge representation methods. This section attempts to provide an overview of these eight categories of models.

2.1. Mathematical definition based model

The mathematical definition based model suggested the use of some mathematical expressions to represent the inner and outer boundaries of each feature (The tolerance information assigned on this feature can be implicitly expressed by this way) [5]. A comprehensive review on mathematical definition based models for the construction of geometric tolerance zones was presented by Pasupathy et al. [6]. The review classified existing mathematical definition based models into offset model, parametric space model, algebraic model, homogeneous transformation model, and parametric curve model. Each category of model has its own advantages and disadvantages. For the details regarding their advantages and disadvantages, please refer to the review in [6].

In addition to the five categories of models described in [6], degrees of freedom model [7–9] and tolerance-map model [10–12] are another two categories of mathematical definition based models. Various TSS design methods based on these two categories of models, such as DOF method [13], functional requirement decomposition method [14], mirror method [15–17], and assembly positioning constraint method [18–20], have been proposed during the past two decades.

2.2. TTRS model

The TTRS model was established by Desrochers and Clement [21]. It firstly organized the functional surfaces of components as TTRS binary trees, then established the minimum geometric datum element (MGDE) of each TTRS, and finally represented tolerances through the MGDEs and the relations between them. The biggest feature of the TTRS model is the reorganization of the geometric information in CAD systems, which can lay a foundation for implementing automatic tolerancing in computer systems.

Based on the TTRS model, Clement et al. [22] presented a method for dimensioning and tolerancing. The presented method firstly extracted the functional surfaces of components and the related relationships between these functional surfaces from CAD system, then used an invariance class [23] and some MGDEs to express each functional surface, and finally generated tolerance types according to the related relationships between MGDEs. To further improve the method, Desrochers and Maranzana [24] designed rules for selecting MGDEs, and Zhang et al. [25] designed a statistical learning based approach for selecting datum reference frames.

2.3. Graph-based model

The graph-based model suggested the use of graphs to represent tolerance information. Typical examples are Tsai and Cutkosky's tolerance network model [26], Hu et al.'s VGC network model [27], and Franciosa et al.'s graph-based model [28, 29]. The tolerance network model applied a uniform graph-based representation scheme, called tolerance network, to represent the tolerance information of a component. The model can accommodate the tolerance specifications related to the function, manufacturing, and inspection requirements and support the use of different types of tolerances. It has been extended to establish a VGC network model by Hu et al. For a simple component or assembly, the extended model can be easily used to generate its TSSs. However, for a complex assembly, the construction process of its graph-based tolerance network or VGC network is time consuming and complicated. The graph-based model represented tolerance information via assembly graph, part graphs, and feature graphs. It was firstly established to validate the global consistency of a 3D TSS set, and was then applied in the automatic calculation of the variational parameters for planar or cylindrical features for a given set of TSSs.

2.4. Modeling language model

The modeling language model suggested the use of some computer readable modeling languages to directly represent tolerance information. Representative examples are EXPRESS language model [30], Rachuri et al.'s unified modeling language (UML) model [31], Zhao et al.'s extensive markup language model [32], and Dantan et al.'s GeoSpelling formal language model [33–35]. These models can provide several ways to express tolerance information in computer readable formats, which makes it easier for computers to process this information. However, there is yet no evidence that they have been used to generate TSSs or tolerance zones.

2.5. Category theory model

The category theory model was firstly presented by Wang et al. [36] to characterize the information of profile surface texture. It has later been applied by Lu et al. [37] to represent the tolerance information in specification and verification via certain categorical constructors, which include category, object, morphism, pull back, product, functor, and natural transformation. The category theory model can provide an abstract representation of tolerance information. By this way, the ambiguous problem caused by describing the information in natural language in technical handbooks and tolerancing standards can be well addressed.

On the basis of Wang et al.'s model and Lu et al.'s model, Xu et al. [38] and Qi et al. [39, 40] further developed knowledge-based systems for the manipulation of tolerance information and surface texture information. The systems enable mechanical designers to query specific rules to design TSSs and surface texture specification schemes, and also offer a structural mapping from a design model to an inspection model. With such mapping, a measurement plan can be automatically generated from its specification requirement.

2.6. Polychromatic set model

The polychromatic set model was constructed by Zhang et al. [41] and can be seen as an extension of the VGC network model [27]. The model was hierarchically organized and consists of an assembly layer, a part layer, an assembly feature surface layer, an assembly tolerance specification layer, and an assembly tolerance zone layer. The relationships in each layer, the VGCs, and the tolerance types were formalized by one or more polychromatic sets. Based on such formalization, the tolerance types for each explicit assembly requirement can be inferred and a graph-based tolerance network can then be established.

On the basis of the polychromatic set model, Zhang et al. [42] designed a reasoning algorithm for the generation of TSSs and corresponding tolerance zone types. This algorithm used a unified formalized model to describe the whole reasoning process from assembly to TSSs and tolerance zone types, which facilitates the realization of the systematization and computerization of TSS and tolerance zone type design.

2.7. Adjacency matrix model

The adjacency matrix model, which was established by Qin et al. [3], is also hierarchically organized. It consists of a component layer, an assembly feature layer, a spatial relation layer, and a tolerance type layer. The relationships in each layer were represented by one or more adjacency matrices. The correspondences between assembly features and tolerance types and between spatial relations and tolerance types were also represented by adjacency matrices. By performing adjacency matrices based inference, tolerance types can be automatically generated. As analyzed in the introduction, the adjacency matrix model is simple, intuitive, and easy to be implemented, but it does not include the representation of VGCs and corresponding tolerance zones and does not consider the generation of the whole TSSs and tolerance zones.

2.8. Ontology-based model

The ontology-based model was firstly presented by Fiorentini et al. [43]. The presented model is actually a web ontology language (OWL) ontology version of Rachuri et al.'s UML model [31]. Due to the rigorous logic-based semantics of OWL, the OWL ontology version of the UML model was expected to be used in semantic interoperability of tolerance information. However, it does not involve the representation of the information of VGCs and tolerance zones because the UML model does not involve it. Inspired by Fiorentini et al.'s work, Lu et al. [44] presented the OWL ontology representation of VGC information, Zhong et al. [45] presented the description logic (the mathematical basis of OWL) representation of assembly tolerance types, and Qin et al. [46, 47] presented the OWL ontology representation of the information of singular tolerances and composite positional tolerance.

In addition to leveraging OWL ontology to represent tolerance information, Zhong et al. [48] presented an OWL ontology-based approach to automatically generate assembly tolerance types in CAD systems. The presented approach firstly used ontology axioms to formalize the relationships in each layer of the adjacency matrix model [3], then used ontology rules to express the mappings from such relationships to tolerance types, and finally implemented automatic generation of tolerance types on the basis of ontology-supported reasoning. The approach was then extended to generate tolerance zones and the whole TSSs by Qin et al. [49, 50].

OWL ontology has advantages in the aspects of conformance checking, computer-interpretable format, semantic representation, and logic-based reasoning [51]. Thus, it comes as no surprise that the use of OWL ontologies in tolerance information representation and TSS design is gaining popularity. However, the OWL ontology-based model is probably not an efficient representation model for tolerance information as a whole, since OWL ontology has drawbacks in the aspects of automation degree, time complexity, additional work, and negation in the model [2].

3. Tolerance representation model

Every assembly consists of a set of components that have mating relations between each other. Every component consists of one or more geometric features. The geometric features of different components of an assembly may have mating relations and VGC relations between each other, where such relations are actually the refinement of the mating relations between components. A tolerance is essentially a VGC between geometric features, while a tolerance zone is the range of the variation of a geometric feature in 3D space which is restricted by the tolerance assigned on this geometric feature. Therefore, from a top-down point of view, the establishment of a tolerance representation model for generating TSSs and tolerance zones should start from formalizing the mating relations between components, and then respectively formalize the mating relations and VGC relations between geometric features. On the basis of such formalizations, the TSSs assigned on geometric features and the tolerance zones of TSSs can be respectively modeled.

Finally, the generation of TSSs and tolerance zones can be implemented based on the formalizations and models.

In this section, a tolerance representation model for generating TSSs and tolerance zones is established. The basic structure of the model is shown in Fig. 1. As can be seen from the figure, the model consists of five layers: component layer, geometric feature layer, VGC layer, TSS layer, and tolerance zone layer. The knowledge in each layer is respectively formalized by one or more adjacency matrices in the section.

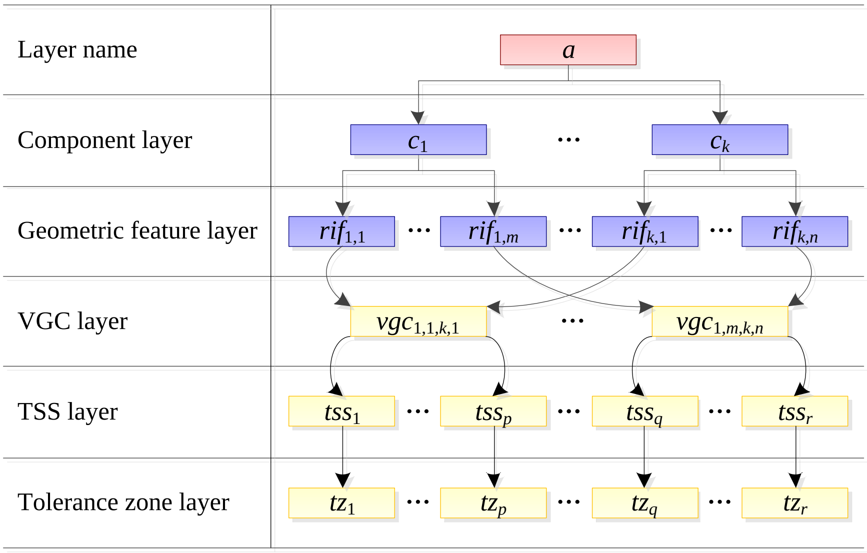


Fig. 1 Basic structure of the tolerance representation model

3.1. Component layer

Component layer is the first layer of the model. Its main functions are to extract the component information and the information of mating relations between components from CAD system, to construct an adjacency matrix for the mating relations between components, and to lay foundation for the latter four layers.

Every assembly can be seen as a set consisting of one or more components which have certain mating relations. To construct an adjacency matrix for the mating relations between the components of an assembly, the component information and the information of mating relations between the components of this assembly need to be extracted firstly. The extracted information is then to be formalized by adjacency matrix.

For instance, the component information of the assembly in Fig. 2 [27] is extracted as: component c_1; component c_2. The information of mating relations between these components is extracted as: mating relations between c_1 and c_2; mating relations between c_2 and c_1. Such information can be described by the following adjacency matrix:

$$\boldsymbol{M}_{\mathrm{C},2\times2}=\begin{bmatrix}0 & 1\\ 1 & 0\end{bmatrix}$$

where the rows of $\boldsymbol{M}_{\mathrm{C},2\times2}$ respectively stands for c_1 and c_2; the columns of $\boldsymbol{M}_{\mathrm{C},2\times2}$ respectively stands for c_1 and c_2; “0” stands for there is no mating relation; and “1” stands for there is a mating relation.

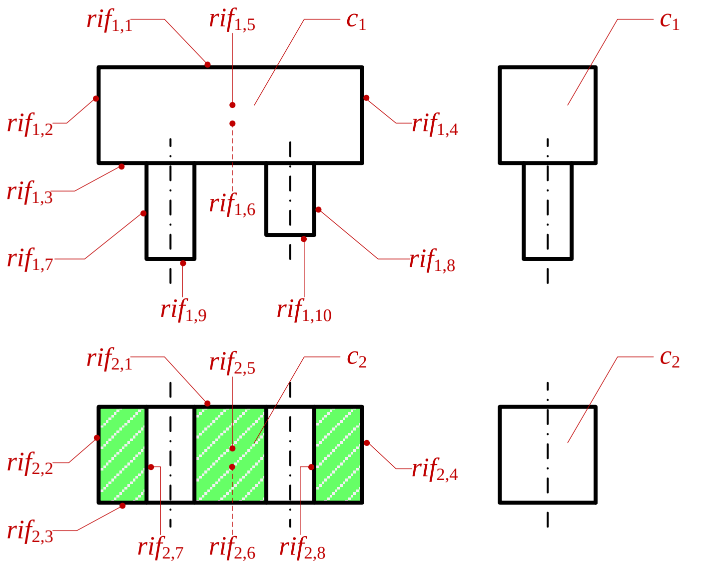


Fig. 2 An assembly and its components and the geometric features forming each component

For an arbitrary assembly, the adjacency matrix for the mating relations between its components can be constructed according to the following definition [3]:

Definition 1 (Adjacency matrix for the mating relations between components). If the given assembly is $a = \{c_1, c_2, \ldots, c_k\}$, where $c_1, c_2, \ldots, c_k$ are the k components that forms a. Then the adjacency matrix for the mating relations between these k components is defined as follow:

$$\boldsymbol{M}_{\mathrm{C},k\times k} = \begin{bmatrix} \alpha_{1,1} & \alpha_{1,2} & \cdots & \alpha_{1,k} \\ \alpha_{2,1} & \alpha_{2,2} & \cdots & \alpha_{2,k} \\ \vdots & \vdots & & \vdots \\ \alpha_{k,1} & \alpha_{k,2} & \cdots & \alpha_{k,k} \end{bmatrix} \quad (1)$$

where the rows of $\boldsymbol{M}_{\mathrm{C},k\times k}$ respectively stands for $c_1, c_2, \ldots, c_k$; the columns of $\boldsymbol{M}_{\mathrm{C},k\times k}$ respectively stands for $c_1, c_2, \ldots, c_k$; if $i = j$ ($i, j = 1, 2, \ldots, k$) or $i \neq j$ and there is no mating relation between c_i and c_j, then $\alpha_{i,j}, \alpha_{j,i} = 0$; and if $i \neq j$ and there is a mating relation between c_i and c_j, then $\alpha_{i,j}, \alpha_{j,i} = 1$.

3.2. Geometric feature layer

Geometric feature layer is the second layer of the model. Its main functions are to extract the information of geometric features and the information of mating relations between geometric features, to construct an adjacency matrix for the mating relations between geometric features, and to lay foundation for the latter three layers.

In feature-based CAD systems, geometric features can be divided into seven invariance classes. They are spherical surface, cylindrical surface, planar surface, helical surface, revolute surface, prismatic surface, and complex surface [23]. Because every component can be seen as a closed geometry surrounded by one or more geometric features that belongs to a specific invariance class, the mating relations between components can be further decomposed into the mating relations between the geometric features of components. To construct an adjacency matrix for the mating relations between geometric features, the geometric feature information of components and the information of mating relations between geometric features are required to be extracted firstly. Then the extracted information is needed to be formalized by adjacency matrix.

For example, the geometric feature information of the components c_1 and c_2 in Fig. 2 is extracted as: geometric features forming c_1 are planar surface $rif_{1,1}$ (*rif* stands for real integral feature), planar surface $rif_{1,2}$, planar surface $rif_{1,3}$, planar surface $rif_{1,4}$, planar surface $rif_{1,5}$, planar surface $rif_{1,6}$, cylindrical surface $rif_{1,7}$, cylindrical surface $rif_{1,8}$, planar surface $rif_{1,9}$, and planar surface $rif_{1,10}$; geometric features forming c_2 are planar surface $rif_{2,1}$, planar surface $rif_{2,2}$, planar surface $rif_{2,3}$, planar surface $rif_{2,4}$, planar surface $rif_{2,5}$, planar surface $rif_{2,6}$, cylindrical surface $rif_{2,7}$, and cylindrical surface $rif_{2,8}$. The information of mating relations between these geometric features is extracted as: mating relations between $rif_{1,7}$ and $rif_{2,7}$; mating relations between $rif_{1,8}$ and $rif_{2,8}$. Such information can be expressed by the following adjacency matrix:

$$\boldsymbol{M}_{\mathbf{GF},2\times 2} = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}$$

where the rows of $\boldsymbol{M}_{\mathbf{GF},2\times 2}$ respectively stands for cylindrical surface $rif_{1,7}$ and cylindrical surface $rif_{1,8}$, and the columns of $\boldsymbol{M}_{\mathbf{C},2\times 2}$ respectively stands for cylindrical surface $rif_{2,7}$ and cylindrical surface $rif_{2,8}$.

For two arbitrary components, the adjacency matrix for the mating relations between their geometric features can be constructed via the following definition:

Definition 2 (Adjacency matrix for the mating relations between geometric features). Let $c_i = \{rif_{i,1}, rif_{i,2}, \ldots, rif_{i,m}\}$ be the *i*-th component of an assembly (where $rif_{i,1}$, $rif_{i,2}$, …, $rif_{i,m}$ are the *m* geometric features that form c_i), and $c_j = \{rif_{j,1}, rif_{j,2}, \ldots, rif_{j,n}\}$ be the *j*-th component of this assembly (where $rif_{j,1}$, $rif_{j,2}$, …, $rif_{j,n}$ are the *n* geometric features that form c_j). If there is a mating relation between c_i and c_j, then the adjacency matrix for the mating relations between their geometric features is defined as follow:

$$\boldsymbol{M}_{\mathbf{GF},m\times n} = \begin{bmatrix} \beta_{1,1} & \beta_{1,2} & \cdots & \beta_{1,n} \\ \beta_{2,1} & \beta_{2,2} & \cdots & \beta_{2,n} \\ \vdots & \vdots & & \vdots \\ \beta_{m,1} & \beta_{m,2} & \cdots & \beta_{m,n} \end{bmatrix} \tag{2}$$

where the rows of $\boldsymbol{M}_{\mathbf{GF},m\times n}$ respectively stands for $rif_{i,1}$, $rif_{i,2}$, …, $rif_{i,m}$; the columns of $\boldsymbol{M}_{\mathbf{GF},m\times n}$ respectively stands for $rif_{j,1}$, $rif_{j,2}$, …, $rif_{j,n}$; if there is no mating relation between $rif_{i,u}$ (u = 1, 2, …, m) and $rif_{j,v}$ (v = 1, 2, …, n) ($rif_{j,v}$ and $rif_{i,u}$), then $\beta_{u,v} = 0$, otherwise $\beta_{u,v} = 1$.

3.3. VGC layer

VGC layer is the third layer of the model. Its main functions are to extract the information of constraints between the geometric features of identical and different components (such information mainly includes the referenced feature, constrained feature, geometric relation, and variational parameter of constraints), to generate the VGCs between geometric features, to construct an adjacency matrix for the VGCs between geometric features, and to lay foundation for the latter two layers.

In feature-based product manufacturing and verification, geometric constraints are essentially the constraints between associated derived feature and real integral feature, the constraints between associated derived features, and the constraints between real integral features. Since such constraints are actually variational, they are called as VGCs. Hu et al. [27] classified VGCs into three categories: (1) Self-referenced VGCs (SVGC). The referenced and constrained features of SVGC are respectively associated derived and real integral features. SVGC totally has seven types: SVGC1, SVGC2, …, SVGC7; (2) Cross-referenced VGCs (CVGC). The referenced and constrained features of CVGC are both associated derived features. CVGC totally has forty-nine types, where twenty-seven ones are called as fundamental CVGCs. They are CVGC1, CVGC2, …, CVGC27; (3) Mating VGCs (MVGC). The referenced and constrained features of MVGC are both real integral features. MVGC totally has forty-nine types, where six ones are called as lower-pair MVGCs. They are MVGC1, MVGC2, …, MVGC6.

To construct an adjacency matrix for the VGCs between geometric features, the first step is to extract the information of constraints between the geometric features of identical and different components. Then the VGC of each constraint is generated in the second step. The final step is to use adjacency matrices to formalize the generated VGCs.

As an example, the information of constraints between the geometric features of the components c_1 and c_2 in Fig. 2 is extracted and listed in Table 1. According to the extracted information and the definition of each type of VGC in [27], the VGC of each constraint in Table 1 is respectively generated and also listed in Table 1. These generated VGCs can be described by the following three adjacency matrices:

$$\boldsymbol{M}_{\mathbf{SVGC},4\times4} = \begin{bmatrix} s_2 & s_0 & s_0 & s_0 \\ s_0 & s_2 & s_0 & s_0 \\ s_0 & s_0 & s_2 & s_0 \\ s_0 & s_0 & s_0 & s_2 \end{bmatrix}$$

$$\boldsymbol{M}_{\mathbf{CVGC},6\times4}=\begin{bmatrix} c_{22} & c_{22} & c_0 & c_0 \\ c_{21} & c_{21} & c_0 & c_0 \\ c_{21} & c_{21} & c_0 & c_0 \\ c_0 & c_0 & c_{22} & c_{22} \\ c_0 & c_0 & c_{21} & c_{21} \\ c_0 & c_0 & c_{21} & c_{21} \end{bmatrix}$$

$$\boldsymbol{M}_{\mathbf{MVGC},2\times2}=\begin{bmatrix} m_2 & m_0 \\ m_0 & m_2 \end{bmatrix}$$

where: (1) The rows of $\boldsymbol{M}_{\mathbf{SVGC},4\times4}$ respectively stands for straight line $adf_{1,7}$, straight line $adf_{1,8}$, straight line $adf_{2,7}$, and straight line $adf_{2,8}$; the columns of $\boldsymbol{M}_{\mathbf{SVGC},4\times4}$ respectively stands for cylindrical surface $rif_{1,7}$, cylindrical surface $rif_{1,8}$, cylindrical surface $rif_{2,7}$, and cylindrical surface $rif_{2,8}$; "s_0" denotes there is no SVGC; and "s_2" denotes SVGC2. (2) The rows of $\boldsymbol{M}_{\mathbf{CVGC},6\times4}$ respectively denotes planar surface $adf_{1,3}$, planar surface $adf_{1,4}$, planar surface $adf_{1,5}$, planar surface $adf_{2,1}$, planar surface $adf_{2,2}$, and planar surface $adf_{2,5}$; the columns of $\boldsymbol{M}_{\mathbf{CVGC},6\times4}$ respectively denotes straight line $adf_{1,7}$, straight line $adf_{1,8}$, straight line $adf_{2,7}$, and straight line $adf_{2,8}$; "c_0" denotes there is no CVGC; "c_{21}" denotes CVGC21; and "c_{22}" denotes CVGC22. (3) The rows of $\boldsymbol{M}_{\mathbf{MVGC},2\times2}$ respectively stands for cylindrical surface $rif_{1,7}$ and cylindrical surface $rif_{1,8}$; the columns of $\boldsymbol{M}_{\mathbf{MVGC},2\times2}$ respectively stands for cylindrical surface $rif_{2,7}$ and cylindrical surface $rif_{2,8}$; "m_0" denotes there is no MVGC; and "m_2" denotes MVGC2.

Table 1 The extracted information of constraints between the geometric features of the components c_1 and c_2 in Fig. 2 and the generated VGCs of these constraints. *adf* stands for associated derived feature. $T(m)$ stands for m (m = 1, 2, 3) independent translations. $R(n)$ stands for n (n = 1, 2, 3) independent rotations. VP denotes variational parameter

Constraint	Referenced feature	Constrained feature	Geometric relation	VP	VGC
Constraint 1	Straight line $adf_{1,7}$	Cylindrical surface $rif_{1,7}$	Constraint relation	$T(2)$, $R(2)$	SVGC2
Constraint 2	Straight line $adf_{1,8}$	Cylindrical surface $rif_{1,8}$	Constraint relation	$T(2)$, $R(2)$	SVGC2
Constraint 3	Straight line $adf_{2,7}$	Cylindrical surface $rif_{2,7}$	Constraint relation	$T(2)$, $R(2)$	SVGC2
Constraint 4	Straight line $adf_{2,8}$	Cylindrical surface $rif_{2,8}$	Constraint relation	$T(2)$, $R(2)$	SVGC2
Constraint 5	Planar surface $adf_{1,3}$	Straight line $adf_{1,7}$	Perpendicular relation	$R(2)$	CVGC22
Constraint 6	Planar surface $adf_{1,4}$	Straight line $adf_{1,7}$	Parallel relation	$T(1)$, $R(1)$	CVGC21
Constraint 7	Planar surface $adf_{1,5}$	Straight line $adf_{1,7}$	Parallel relation	$T(1)$, $R(1)$	CVGC21
Constraint 8	Planar surface $adf_{1,3}$	Straight line $adf_{1,8}$	Perpendicular relation	$R(2)$	CVGC22
Constraint 9	Planar surface $adf_{1,4}$	Straight line $adf_{1,8}$	Parallel relation	$T(1)$, $R(1)$	CVGC21
Constraint 10	Planar surface $adf_{1,5}$	Straight line $adf_{1,8}$	Parallel relation	$T(1)$, $R(1)$	CVGC21
Constraint 11	Planar surface $adf_{2,1}$	Straight line $adf_{2,7}$	Perpendicular relation	$R(2)$	CVGC22
Constraint 12	Planar surface $adf_{2,2}$	Straight line $adf_{2,7}$	Parallel relation	$T(1)$, $R(1)$	CVGC21
Constraint 13	Planar surface $adf_{2,5}$	Straight line $adf_{2,7}$	Parallel relation	$T(1)$, $R(1)$	CVGC21
Constraint 14	Planar surface $adf_{2,1}$	Straight line $adf_{2,8}$	Perpendicular relation	$R(2)$	CVGC22
Constraint 15	Planar surface $adf_{2,2}$	Straight line $adf_{2,8}$	Parallel relation	$T(1)$, $R(1)$	CVGC21
Constraint 16	Planar surface $adf_{2,5}$	Straight line $adf_{2,8}$	Parallel relation	$T(1)$, $R(1)$	CVGC21

Constraint 17	Cylindrical surface $rif_{1,7}$	Cylindrical surface $rif_{2,7}$	Mating relation	$T(1)$, $R(1)$	MVGC2
Constraint 18	Cylindrical surface $rif_{1,8}$	Cylindrical surface $rif_{2,8}$	Mating relation	$T(1)$, $R(1)$	MVGC2

For two arbitrary components, the adjacency matrix for the VGCs between their geometric features can be constructed according to the following definition:

Definition 3 (Adjacency matrices for the VGCs between geometric features). Let $c_i = \{rif_{i,1}, rif_{i,2}, \ldots, rif_{i,m}\}$ be the i-th component of an assembly (where $rif_{i,1}, rif_{i,2}, \ldots, rif_{i,m}$ are the m geometric features that form c_i), $c_j = \{rif_{j,1}, rif_{j,2}, \ldots, rif_{j,n}\}$ be the j-th component of this assembly (where $rif_{j,1}, rif_{j,2}, \ldots, rif_{j,n}$ are the n geometric features that form c_j), $adf_{i,1}, adf_{i,2}, \ldots, adf_{i,m}$ respectively be the associated derived features of the real integral features $rif_{i,1}, rif_{i,2}, \ldots, rif_{i,m}$, and $adf_{j,1}, adf_{j,2}, \ldots, adf_{j,n}$ respectively be the associated derived features of the real integral features $rif_{j,1}, rif_{j,2}, \ldots, rif_{j,n}$. If there is a mating relation between c_i and c_j, then the adjacency matrices for the VGCs between their geometric features are defined as follows:

$$\boldsymbol{M}_{\mathbf{SVGC},(m+n)\times(m+n)} = \begin{bmatrix} \gamma_{1,1} & \gamma_{1,2} & \cdots & \gamma_{1,m+n} \\ \gamma_{2,1} & \gamma_{2,2} & \cdots & \gamma_{2,m+n} \\ \vdots & \vdots & & \vdots \\ \gamma_{m+n,1} & \gamma_{m+n,2} & \cdots & \gamma_{m+n,m+n} \end{bmatrix} \tag{3}$$

$$\boldsymbol{M}_{\mathbf{CVGC},(m+n)\times(m+n)} = \begin{bmatrix} \phi_{1,1} & \phi_{1,2} & \cdots & \phi_{1,m+n} \\ \phi_{2,1} & \phi_{2,2} & \cdots & \phi_{2,m+n} \\ \vdots & \vdots & & \vdots \\ \phi_{m+n,1} & \phi_{m+n,2} & \cdots & \phi_{m+n,m+n} \end{bmatrix} \tag{4}$$

$$\boldsymbol{M}_{\mathbf{MVGC},m\times n} = \begin{bmatrix} \varphi_{1,1} & \varphi_{1,2} & \cdots & \varphi_{1,n} \\ \varphi_{2,1} & \varphi_{2,2} & \cdots & \varphi_{2,n} \\ \vdots & \vdots & & \vdots \\ \varphi_{m,1} & \varphi_{m,2} & \cdots & \varphi_{m,n} \end{bmatrix} \tag{5}$$

where: (1) The rows of $\boldsymbol{M}_{\mathbf{SVGC},(m+n)\times(m+n)}$ respectively stands for $adf_{i,1}, adf_{i,2}, \ldots, adf_{i,m}, adf_{j,1}, adf_{j,2}, \ldots, adf_{j,n}$; the columns of $\boldsymbol{M}_{\mathbf{SVGC},(m+n)\times(m+n)}$ respectively stands for $rif_{i,1}, rif_{i,2}, \ldots, rif_{i,m}, rif_{j,1}, rif_{j,2}, \ldots, rif_{j,n}$; if there is a SVGC$x$ (x = 1, 2, …, 7) between $adf_{k,r}$ ($k = i, j$; when $k = i$, r = 1, 2, …, m; when $k = j$, r = 1, 2, …, n) and $rif_{k,r}$, then: 1) when $k = i$, let $\gamma_{r,r} = \mathrm{s}_x$, 2) when $k = j$, let $\gamma_{(m+r),(m+r)} = \mathrm{s}_x$, and 3) let the values of the remaining elements of $\boldsymbol{M}_{\mathbf{SVGC},(m+n)\times(m+n)}$ be s_0. (2) The rows of $\boldsymbol{M}_{\mathbf{CVGC},(m+n)\times(m+n)}$ respectively denotes $adf_{i,1}, adf_{i,2}, \ldots, adf_{i,m}, adf_{j,1}, adf_{j,2}, \ldots, adf_{j,n}$; the columns of $\boldsymbol{M}_{\mathbf{CVGC},(m+n)\times(m+n)}$ respectively denotes $adf_{i,1}, adf_{i,2}, \ldots, adf_{i,m}, adf_{j,1}, adf_{j,2}, \ldots, adf_{j,n}$; if there is no CVGC between $adf_{p,s}$ ($p = i, j$; when $p = i$, s = 1, 2, …, m; when $p = j$, s = 1, 2, …, n) and $adf_{q,t}$ ($q = i, j$; when $q = i$, t = 1, 2, …, m; when $q = j$, t = 1, 2, …, n), then: 1) when $p = i$ and $q = i$, let $\phi_{s,t} = \mathrm{c}_0$, 2) when $p = i$ and $q = j$, let $\phi_{s,m+t} = \mathrm{c}_0$, 3) when $p = j$ and $q = i$, let $\phi_{m+s,t} = \mathrm{c}_0$, and 4) when $p = j$ and $q = j$, let $\phi_{m+s,m+t} = \mathrm{c}_0$; if there is a CVGC$y$ (y = 1, 2, …, 27) between $adf_{p,s}$ and $adf_{q,t}$, then: 1) when p =

i and $q = i$, let $\phi_{s,t} = c_y$, 2) when $p = i$ and $q = j$, let $\phi_{s,m+t} = c_y$, 3) when $p = j$ and $q = i$, let $\phi_{m+s,t} = c_y$, and 4) when $p = j$ and $q = j$, let $\phi_{m+s,m+t} = c_y$. (3) The rows of $\boldsymbol{M}_{\mathbf{MVGC},m\times n}$ respectively stands for $rif_{i,1}$, $rif_{i,2}$, …, $rif_{i,m}$; The columns of $\boldsymbol{M}_{\mathbf{MVGC},m\times n}$ respectively stands for $rif_{j,1}$, $rif_{j,2}$, …, $rif_{j,n}$; if there is no MVGC between $rif_{i,u}$ (u = 1, 2, …, m) and $rif_{j,v}$ (v = 1, 2, …, n), then $\varphi_{u,v} = m_0$; if there is a MVGCz (z = 1, 2, …, 6) between $rif_{i,u}$ and $rif_{j,v}$, then $\varphi_{u,v} = m_z$.

3.4. TSS layer

TSS layer is the fourth layer of the model. Its main functions are to generate the TSSs of each component (the details will be explained in Section 4), to construct an adjacency matrix for the TSSs of each component, and to lay foundation for the last layer.

A TSS consists of four parts: datum system (if required), toleranced feature, tolerance type, and tolerance principle (if required). Its framework is shown in Fig. 3. A datum is an ideal feature (an ideal point, straight line, or plane) that is used to constrain the direction and position of toleranced feature. A datum system consists of two or three datums that are in a prioritized order. When it includes two datums, the two datums are respectively called primary datum and secondary datum. When it includes three datums, the three datums are respectively called primary datum, secondary datum, and tertiary datum. A datum or a datum system is required in orientation, location, and run-out tolerances. A TSS may include no datum, an individual datum, or a datum system.

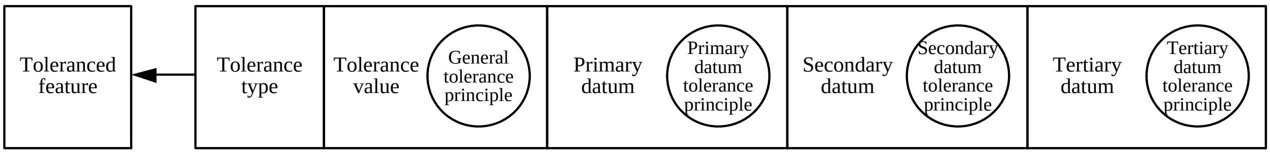


Fig. 3 The framework of a TSS

Toleranced feature is the feature that is applied a TSS. It can be classified into four categories of point, line, surface, and feature of size. Surface includes spherical surface, cylindrical surface, planar surface, helical surface, revolute surface, prismatic surface, and complex surface. Feature of size is the derived feature of the feature whose size is determined by dimension. It mainly includes center of sphere, axis of cylinder, and mid-plane of slot.

Tolerance type is used to describe the geometric characteristics of feature. It mainly incudes: (1) linear dimension tolerance and angle tolerance that are used to describe the dimensional characteristic of feature; (2) straightness, flatness, roundness, cylindricity, profile any line, and profile any surface that are used to describe the form characteristic of feature; (3) parallelism, perpendicularity, angularity, profile any line, and profile any surface that are used to describe the orientation characteristic of feature; (4) position, concentricity, coaxiality, symmetry, profile any line, and profile any surface that are used to describe the location characteristic of feature; (5) circular run-out and total run-out that are used to describe the run-out characteristic of feature.

Tolerance principle is used to express the intrinsic relationship between dimension and geometric tolerances. It can be divided into two categories of independent principle and interrelated requirements, where interrelated requirements include envelope requirement (Ⓔ), maximum material requirement (Ⓜ), least material requirement (Ⓛ), and reciprocity requirement (Ⓡ). If there is no intrinsic relationship between the designed dimension and geometric tolerances, independent principle will be applied. Otherwise, envelope requirement, maximum material requirement, least material requirement, or reciprocity requirement may be used. Tolerance principles can be applied on the whole tolerance, the primary datum, the secondary datum, and the tertiary datum. They are called general tolerance principle, primary datum tolerance principle, secondary datum tolerance principle, and tertiary datum tolerance principle, respectively.

To construct an adjacency matrix for the TSSs of each component, such TSSs are required to be generated firstly. For instance, assume the generated TSSs of the component c_2 in Fig. 2 are the TSSs in Fig. 4. Then these TSSs can be expressed by the following adjacency matrix:

$$\boldsymbol{M}_{\mathbf{TSS},3\times10} = \begin{bmatrix} \mathrm{A}(rif_{2,7}) & \mathrm{tt}_{12} & \phi t_1 & 0 & \mathrm{A} & 0 & \mathrm{B} & 0 & \mathrm{C} & 0 \\ \mathrm{A}(rif_{2,8}) & \mathrm{tt}_{12} & \phi t_2 & 0 & \mathrm{A} & 0 & \mathrm{B} & 0 & \mathrm{C} & 0 \\ rif_{2,3} & \mathrm{tt}_4 & t_3 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

where the rows of $\boldsymbol{M}_{\mathbf{TSS},3\times10}$ respectively stands for TSS tss_1 whose tolerance value is ϕt_1, TSS tss_2 whose tolerance value is ϕt_2, and TSS tss_3 whose tolerance value is t_3; the columns of $\boldsymbol{M}_{\mathbf{TSS},3\times10}$ respectively stands for toleranced feature, tolerance type, tolerance value, general tolerance principle, primary datum, primary datum tolerance principle, secondary datum, secondary datum tolerance principle, tertiary datum, and tertiary datum tolerance principle; "A($rif_{2,7}$)" denotes the axis of the cylindrical surface $rif_{2,7}$; "A($rif_{2,8}$)" denotes the axis of the cylindrical surface $rif_{2,8}$; "tt_4" denotes flatness; "tt_{12}" denotes position; and "0" denotes null.

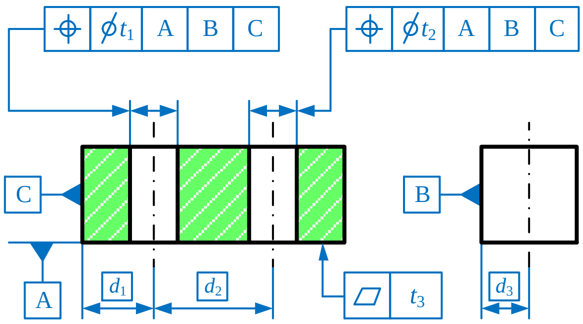


Fig. 4 The generated TSSs of the component c_2 in Fig. 2

For an arbitrary component, the adjacency matrix for its TSSs can be constructed according to the following definition:

Definition 4 (Adjacency matrix for the TSSs of component). Let $c_i = \{rif_{i,1}, rif_{i,2}, \ldots, rif_{i,m}\}$ be the i-th component of an assembly (where $rif_{i,1}, rif_{i,2}, \ldots, rif_{i,m}$ are the m geometric features that form c_i), and $fos_{i,1}$, $fos_{i,2}, \ldots, fos_{i,m}$ be the features of size of $rif_{i,1}, rif_{i,2}, \ldots, rif_{i,m}$, respectively (If $rif_{i,u}$ ($u = 1, 2, \ldots, m$) does not have feature of size, then $fos_{i,u}$ is null). Then the adjacency matrix for the TSSs of c_i is defined as follow:

$$\boldsymbol{M}_{\mathbf{TSS},r\times 10} = \begin{bmatrix} \xi_{1,1} & \xi_{1,2} & \xi_{1,3} & \xi_{1,4} & \xi_{1,5} & \xi_{1,6} & \xi_{1,7} & \xi_{1,8} & \xi_{1,9} & \xi_{1,10} \\ \xi_{2,1} & \xi_{2,2} & \xi_{2,3} & \xi_{2,4} & \xi_{2,5} & \xi_{2,6} & \xi_{2,7} & \xi_{2,8} & \xi_{2,9} & \xi_{2,10} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \xi_{r,1} & \xi_{r,2} & \xi_{r,3} & \xi_{r,4} & \xi_{r,5} & \xi_{r,6} & \xi_{r,7} & \xi_{r,8} & \xi_{r,9} & \xi_{r,10} \end{bmatrix} \tag{6}$$

where the rows of $\boldsymbol{M}_{\mathbf{TSS},r\times 10}$ respectively denotes TSS tss_1 whose tolerance value is $(S)(\phi)t_1$, TSS tss_2 whose tolerance value is $(S)(\phi)t_2$, …, TSS tss_r whose tolerance value is $(S)(\phi)t_r$; the columns of $\boldsymbol{M}_{\mathbf{TSS},r\times 10}$ respectively denotes toleranced feature, tolerance type, tolerance value, general tolerance principle, primary datum, primary datum tolerance principle, secondary datum, secondary datum tolerance principle, tertiary datum, and tertiary datum tolerance principle; $\xi_{1,1}, \xi_{2,1}, \ldots, \xi_{r,1} \in \{rif_{i,1}, rif_{i,2}, \ldots, rif_{i,m}, fos_{i,1}, fos_{i,2}, \ldots, fos_{i,m}\}$; $\xi_{1,2}$, $\xi_{2,2}, \ldots, \xi_{r,2} \in \{\mathrm{tt}_1, \mathrm{tt}_2, \ldots, \mathrm{tt}_{17}\}$ (tt_1 denotes linear dimension tolerance; tt_2 denotes angle tolerance; tt_3 denotes straightness; tt_4 denotes flatness; tt_5 denotes roundness; tt_6 denotes cylindricity; tt_7 denotes profile any line; tt_8 denotes profile any surface; tt_9 denotes parallelism; tt_{10} denotes perpendicularity; tt_{11} denotes angularity; tt_{12} denotes position; tt_{13} denotes concentricity; tt_{14} denotes coaxiality; tt_{15} denotes symmetry; tt_{16} denotes circular run-out; tt_{17} denotes total run-out); $\xi_{1,3}, \xi_{2,3}, \ldots, \xi_{r,3} \in \{(S)(\phi)t_1, (S)(\phi)t_2, \ldots, (S)(\phi)t_r\}$; $\xi_{1,4}, \xi_{2,4}, \ldots$, $\xi_{r,4}$∈{Ⓔ,Ⓜ,Ⓛ,Ⓡ}; $\xi_{1,5}, \xi_{2,5}, \ldots, \xi_{r,5} \in \{\mathrm{A}, \mathrm{B}, \mathrm{C}\}$; $\xi_{1,6}, \xi_{2,6}, \ldots, \xi_{r,6}$∈{Ⓔ,Ⓜ,Ⓛ,Ⓡ}; $\xi_{1,7}, \xi_{2,7}, \ldots, \xi_{r,7} \in \{\mathrm{A}, \mathrm{B}, \mathrm{C}\}$; $\xi_{1,8}, \xi_{2,8}, \ldots, \xi_{r,8}$∈{Ⓔ,Ⓜ,Ⓛ,Ⓡ}; $\xi_{1,9}, \xi_{2,9}, \ldots, \xi_{r,9} \in \{\mathrm{A}, \mathrm{B}, \mathrm{C}\}$; and $\xi_{1,10}, \xi_{2,10}, \ldots, \xi_{r,10}$∈{Ⓔ,Ⓜ,Ⓛ,Ⓡ}.

3.5. Tolerance zone layer

Tolerance zone layer is the last layer of the model. Its main functions are to generate the tolerance zone of each TSS (the details will be explained in Section 4), to construct an adjacency matrix for the tolerance zones of component TSSs, and to aid further tolerance allocation and analysis.

Tolerance* assigned on toleranced feature restricts the geometric variation of tolerance feature in 3D space. The range of such variation is tolerance zone. Tolerance zone has four basic attributes: form, size, orientation, and location. As depicted in Fig. 5, there are totally ten different tolerance zone forms. They are circle, sphere, cylinder, two coaxial cylinders, two concentric circles, two parallel circles, two parallel straight lines, two parallel planes, two enveloping curves, and two enveloping surfaces. Tolerance zone size is its width or diameter, which is commonly referred to as the tolerance value. Tolerance zone orientation and location are determined by their corresponding tolerance: (1) Both the tolerance zone orientation and location of form tolerances are unrestricted. (2) The tolerance zone orientation of orientation tolerances is restricted, but their tolerance zone location is unrestricted. (3) Both the tolerance zone orientation and location of location and run-out tolerances are restricted.

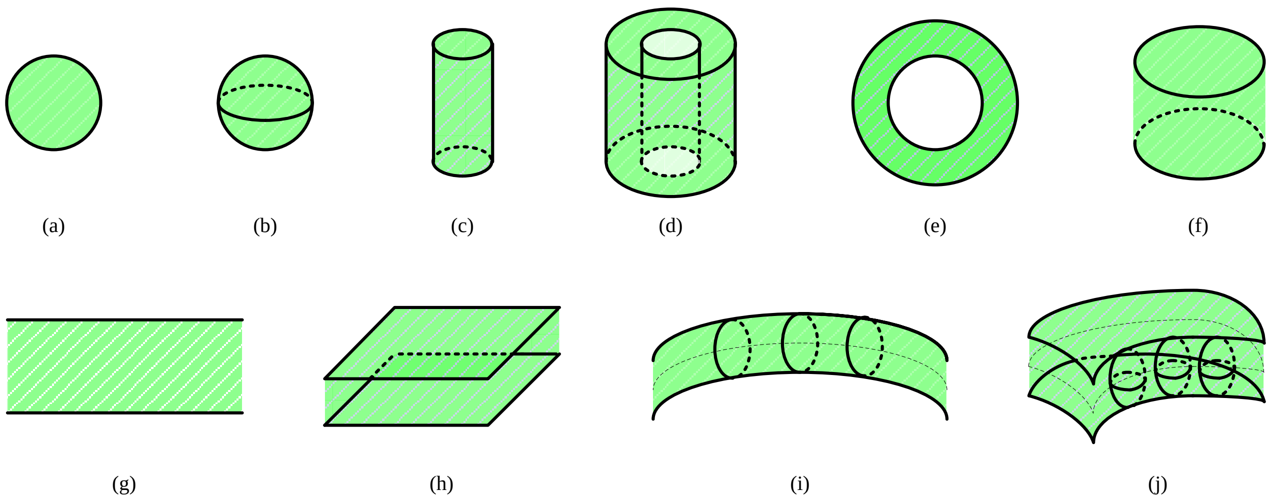


Fig. 5 Ten different tolerance zone forms

To construct an adjacency matrix for the tolerance zones of component TSSs, such tolerance zones are required to be generated firstly. Then the generated tolerance zones are needed to be formalized using adjacency matrix. For example, assume the generated tolerance zones of the component TSSs in Fig. 4 are the tolerance zones in Fig. 6. Then these tolerance zones can be described by the following adjacency matrix:

$$\boldsymbol{M}_{\mathbf{TZ},3\times4}=\begin{bmatrix} \mathrm{c} & \phi t_1 & \mathrm{O}_1 & \mathrm{L}_1 \\ \mathrm{c} & \phi t_2 & \mathrm{O}_2 & \mathrm{L}_2 \\ \mathrm{h} & t_3 & 0 & 0 \end{bmatrix}$$

*Tolerance in this paper refers to a general tolerance (i.e. a single tolerance) and does not include composite tolerances because the representation of the tolerance zones of composite tolerances requires a series of complicated handlings and is beyond the major scope of the paper. For the details regarding this, please refer to [4, 47, 52].

where the rows of $\boldsymbol{M}_{\mathbf{TZ},3\times4}$ respectively stands for the tolerance zone of tss_1 (tz_1), the tolerance zone of tss_2 (tz_2), and the tolerance zone of tss_3 (tz_3); the columns of $\boldsymbol{M}_{\mathbf{TZ},3\times4}$ respectively stands for form, size, orientation, and location; "c" denotes cylinder; "h" denotes two parallel planes; "O_1" denotes tz_1 is perpendicular to A; "O_2" denotes tz_2 is perpendicular to A; "L_1" denotes the distance from the axis of tz_1 to B is theoretically exact dimension d_3 and the distance from the axis of tz_1 to C is theoretically exact dimension d_1; "L_2" denotes the distance from the axis of tz_2 to B is theoretically exact dimension d_3 and the distance from the axis of tz_2 to C is theoretically exact dimension d_1+d_2; and "0" denotes null.

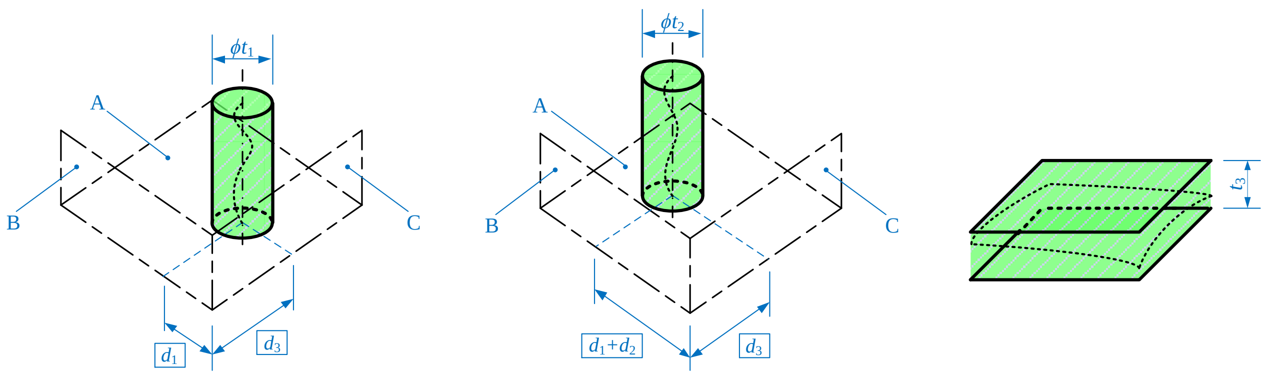


Fig. 6 The generated tolerance zones of the component TSSs in Fig. 4

For an arbitrary component, the adjacency matrix for the tolerance zones of its TSSs can be constructed according to the following definition:

Definition 5 (Adjacency matrix for the tolerance zones of component TSSs). If the tolerance specification schemes designed on a component are tss_1, tss_2, …, tss_r, and $(S)(\phi)t_1$, $(S)(\phi)t_2$, …, $(S)(\phi)t_r$ are respectively the tolerance values in tss_1, tss_2, …, tss_r, then the adjacency matrix for the tolerance zones of the TSSs of this component is defined as follow:

$$\boldsymbol{M}_{\mathbf{TZ},r\times4}=\begin{bmatrix}\zeta_{1,1} & \zeta_{1,2} & \zeta_{1,3} & \zeta_{1,4}\\ \zeta_{2,1} & \zeta_{2,2} & \zeta_{2,3} & \zeta_{2,4}\\ \vdots & \vdots & \vdots & \vdots\\ \zeta_{r,1} & \zeta_{r,2} & \zeta_{r,3} & \zeta_{r,4}\end{bmatrix} \tag{7}$$

where the rows of $\boldsymbol{M}_{\mathbf{TZ},r\times4}$ respectively denotes the tolerance zone of tss_1 (tz_1), the tolerance zone of tss_2 (tz_2), …, the tolerance zone of tss_r (tz_r); the columns of $\boldsymbol{M}_{\mathbf{TZ},r\times4}$ respectively stands for form, size, orientation, and location; $\zeta_{1,1}$, $\zeta_{2,1}$, …, $\zeta_{r,1}\in\{$a, b, …, j$\}$ (a denotes circle, b denotes sphere, c denotes cylinder, d denotes two coaxial cylinders, e denotes two concentric circles, f denotes two parallel circles, g denotes two parallel straight lines, h denotes two parallel planes, i denotes two enveloping curves, j denotes two enveloping surfaces); $\zeta_{1,2}$, $\zeta_{2,2}$, …, $\zeta_{r,2}\in\{(S)(\phi)t_1, (S)(\phi)t_2,…, (S)(\phi)t_r\}$; $\zeta_{1,3}$, $\zeta_{2,3}$, …, $\zeta_{r,3}\in\{O_1, O_2, …, O_r\}$ (O_1, O_2, …, O_r are the orientation of tz_1, tz_2, …, tz_r, respectively); $\zeta_{1,4}$, $\zeta_{2,4}$, …, $\zeta_{r,4} \in \{L_1, L_2, …, L_r\}$ (L_1, L_2, …, L_r are the location of tz_1, tz_2, …, tz_r, respectively).

4. Generation method

This section describes a generation method of TSSs and corresponding tolerance zones based on the proposed tolerance representation model in Section 3. The schematic representation of this method is shown in Fig. 7. As depicted in the figure, to generate the TSSs of each component of an assembly and the tolerance zones of TSSs, the information is firstly extracted from the assembly. Then $\boldsymbol{M}_{\mathbf{C},k\times k}$ and $\boldsymbol{M}_{\mathbf{GF},m\times n}$ are respectively constructed according to the extracted information. Based on $\boldsymbol{M}_{\mathbf{C},k\times k}$ and $\boldsymbol{M}_{\mathbf{GF},m\times n}$, the VGCs between geometric features are generated. $\boldsymbol{M}_{\mathbf{SVGC},(m+n)\times(m+n)}$, $\boldsymbol{M}_{\mathbf{CVGC},(m+n)\times(m+n)}$, and $\boldsymbol{M}_{\mathbf{MVGC},m\times n}$ are constructed on the basis of the generated VGCs. According to $\boldsymbol{M}_{\mathbf{SVGC},(m+n)\times(m+n)}$, $\boldsymbol{M}_{\mathbf{CVGC},(m+n)\times(m+n)}$, $\boldsymbol{M}_{\mathbf{MVGC},m\times n}$, and the procedures of selecting datum, identifying toleranced features, generating tolerance types, and applying tolerance principles, the TSSs of each component are generated and $\boldsymbol{M}_{\mathbf{TSS},r\times 10}$ is thus constructed. Finally, the tolerance zone of each TSS is generated according to $\boldsymbol{M}_{\mathbf{TSS},r\times 10}$ and $\boldsymbol{M}_{\mathbf{TZ},r\times 4}$ is therefore constructed. It can be concluded from these descriptions that the general generation process is: Extract information → Construct $\boldsymbol{M}_{\mathbf{C},k\times k}$ and $\boldsymbol{M}_{\mathbf{GF},m\times n}$ → Generate VGCs → Construct $\boldsymbol{M}_{\mathbf{SVGC},(m+n)\times(m+n)}$, $\boldsymbol{M}_{\mathbf{CVGC},(m+n)\times(m+n)}$, and $\boldsymbol{M}_{\mathbf{MVGC},m\times n}$ → Generate TSSs → Construct $\boldsymbol{M}_{\mathbf{TSS},r\times 10}$ → Generate tolerance zones → Construct $\boldsymbol{M}_{\mathbf{TZ},r\times 4}$. The details of this process are explained in the following steps:

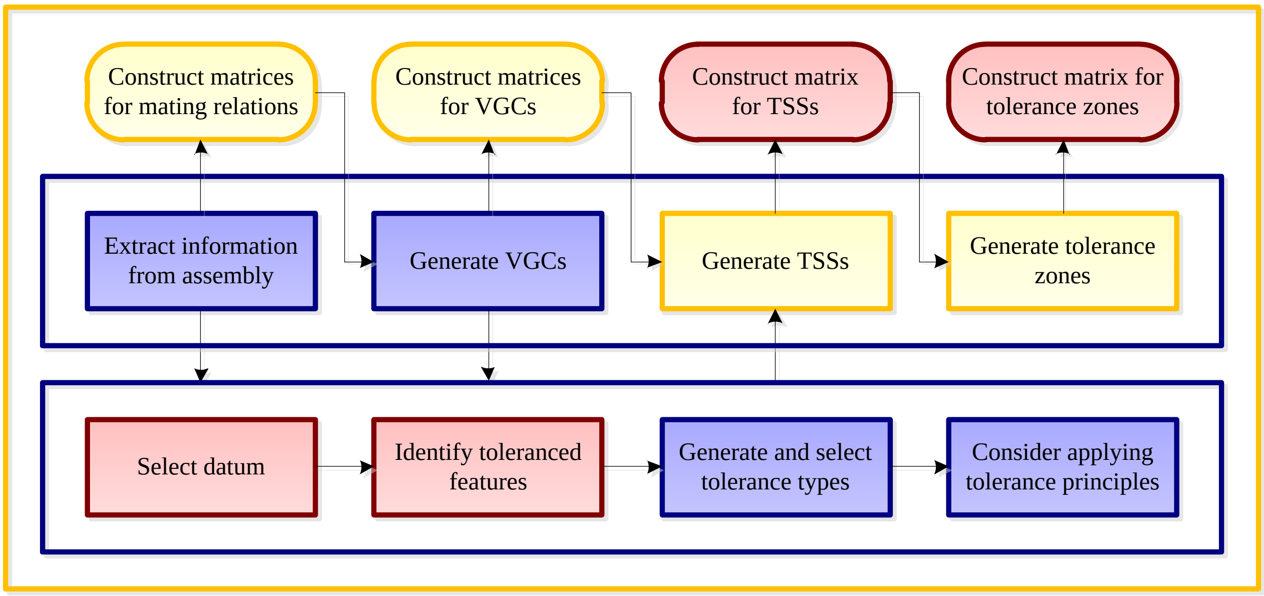


Fig. 7 Schematic representation of the generation method

- Extract information from assembly. Using the application programming interface (API) of a CAD system, the information of components, mating relations between components, geometric features, mating relations between geometric features, and constraints between geometric features is extracted from the assembly that is designed in this CAD system.

- Construct matrices for mating relations. According to the extracted information of components, mating relations between components, geometric features, mating relations between geometric features, Definition 1, and Definition 2, an adjacency matrix for the mating relations between components ($\boldsymbol{M}_{\mathbf{C},k\times k}$) and an adjacency matrix for the mating relations between geometric features ($\boldsymbol{M}_{\mathbf{GF},m\times n}$) are respectively constructed.
- Generate VGCs. On the basis of the constructed adjacency matrices $\boldsymbol{M}_{\mathbf{C},k\times k}$ and $\boldsymbol{M}_{\mathbf{GF},m\times n}$ and the definition of each type of VGC in [27], the VGCs between geometric features are generated.
- Construct matrices for VGCs. According to the generated VGCs and Definition 3, three adjacency matrices for the VGCs between geometric features ($\boldsymbol{M}_{\mathbf{SVGC},(m+n)\times(m+n)}$, $\boldsymbol{M}_{\mathbf{CVGC},(m+n)\times(m+n)}$, and $\boldsymbol{M}_{\mathbf{MVGC},m\times n}$) are constructed.
- Select datum or datum system. There are a number of existing methods that can be used to guide datum or datum system selection. Representative methods are Ballu and Mathieu's method [14], Wu et al.'s method [13], Wang et al.'s method [15], Anselmetti's method [19], Zhang et al.'s method [25], and Armillotta's method [16]. On the basis of Wang et al.'s method [15], this step selects datum or datum system by the following ways: (1) Extract mating features (i.e. the geometric features which actually participate in mates) and relations from the constructed adjacency matrix $\boldsymbol{M}_{\mathbf{GF},m\times n}$. (2) Determine the actual function of the mate in each mating relation. According to Wang et al.'s method, the actual functions of mates can be classified into location, seat, contact, and alignment functions. On the basis of such classification, the actual function of the mate in each mating relation is determined. (3) Determine datum or datum system. 1) If a geometric feature participates in a mate that has location function, it is selected as the candidate feature for secondary datum feature. 2) If a geometric feature participates in a mate that has seat function, it is selected as the candidate feature for primary datum feature. 3) If a geometric feature participates in a mate that has contact function, it is selected as the candidate feature for tertiary datum feature. 4) If a geometric feature participates in a mate that has alignment function, it is selected as the candidate feature for secondary datum feature. 5) If the current component is a hole component or a shaft component, its axis or the common axis of its two geometric features is selected as the candidate feature for primary datum feature. 6) Synthesizing the candidate datum features that are selected by these rules, the datum or datum system of a component can be determined.
- Identify toleranced features. For a concrete component, its toleranced features are identified as the re-

maining features obtained from removing all its datum features from all its mating features. In addition, the features of size of these toleranced features can also be selected as candidate toleranced features (For a concrete feature of size, whether it is selected as a toleranced feature depends on its actual function).

- Generate and select tolerance types. According to the constructed adjacency matrices $\boldsymbol{M}_{\mathbf{SVGC},(m+n)\times(m+n)}$, $\boldsymbol{M}_{\mathbf{CVGC},(m+n)\times(m+n)}$, and $\boldsymbol{M}_{\mathbf{MVGC},m\times n}$ and the mapping relations from each type of VGC to tolerance types in [27], the possible tolerance types for each toleranced feature are generated. Then the final tolerance types for this toleranced feature are specified from the generated possible tolerance types according to the actual function of the toleranced feature.
- Consider applying tolerance principles. According to the specified tolerance types for each toleranced feature and the actual function of this toleranced feature, whether a general tolerance principle or a datum tolerance principle is required to be applied to a designed tolerance type can be determined. The following five rules, which are summarized from [15], are usually used to guide the application of tolerance principles: (1) If there is no intrinsic relationship between the designed dimension and geometric tolerances, an independent principle will be applied by default. (2) If a toleranced feature is used to maintain the symmetry between two components and is selected as a datum feature, then it cannot be applied a maximum material requirement. (3) If a toleranced feature is used to connect two parts, then it needs to be applied a maximum material requirement. (4) If a toleranced feature is in an interference fit, then a regardless of feature size is recommended for it. (5) If a toleranced feature is in a clearance fit, then a maximum material requirement is recommended for it.
- Generate TSSs. On the basis of the selected datum, identified toleranced features, specified tolerance types, and applied tolerance principles (if required), the TSSs for each component are generated.
- Construct matrix for TSSs. According to the generated TSSs for each component and Definition 4, an adjacency matrix for the TSSs of this component ($\boldsymbol{M}_{\mathbf{TSS},r\times 10}$) is constructed.
- Generate tolerance zones. According to the constructed adjacency matrix $\boldsymbol{M}_{\mathbf{TSS},r\times 10}$, the concrete tolerance type (Table 2 lists thirty-eight different concrete tolerance types) in each TSS is determined. Then the tolerance zone form of this TSS can be obtained on the basis of the correspondences between concrete tolerance types and tolerance zone forms in Table 2. The tolerance zone of the TSS is generated according to the obtained form and the orientation and location of the TSS in $\boldsymbol{M}_{\mathbf{TSS},r\times 10}$.

Table 2 Thirty-eight different concrete tolerance types and the correspondences between these concrete tolerance types and the ten different tolerance zone forms in Fig. 5. CTT denotes concrete tolerance type

Tolerance type	Concrete tolerance type	(a)	(b)	(c)	(d)	(e)	(f)	(g)	(h)	(i)	(j)

Straightness	CTT1: On a given plane							√			
	CTT2: On a given orientation								√		
	CTT3: On arbitrary orientation			√							
Flatness	CTT4								√		
Roundness	CTT5					√					
Cylindricity	CTT6				√						
Profile any line	CTT7									√	
Profile any surface	CTT8										√
Parallelism	CTT9: Line to datum system								√		
	CTT10: Line to datum Line			√							
	CTT11: Line to datum plane								√		
	CTT12: Plane to datum line								√		
	CTT13: Plane to datum plane								√		
Perpendicularity	CTT14: Line to datum system								√		
	CTT15: Line to datum Line								√		
	CTT16: Line to datum plane			√							
	CTT17: Plane to datum line								√		
	CTT18: Plane to datum plane								√		
Angularity	CTT19: Line to datum Line								√		
	CTT20: Line to datum plane								√		
	CTT21: Plane to datum line								√		
	CTT22: Plane to datum plane								√		
Position	CTT23: Point		√								
	CTT24: Line on a given orientation								√		
	CTT25: Line on two given orientations								√		
	CTT26: Line on arbitrary orientation			√							
	CTT27: Profile plane								√		
Concentricity	CTT28	√									
Coaxiality	CTT29			√							
Symmetry	CTT30								√		
Profile any line	CTT31									√	
Profile any surface	CTT32										√
Circular run-out	CTT33: On radial orientation					√					
	CTT34: On end surface						√				
	CTT35: On inclined face						√				
	CTT36: On inclined end surface						√				
Total run-out	CTT37: On radial orientation				√						
	CTT38: On end surface								√		

- Construct matrix for tolerance zones. According to the generated tolerance zones of the TSSs of each component and Definition 5, an adjacency matrix for these tolerance zones ($\boldsymbol{M}_{\mathbf{TZ},r\times 4}$) is constructed.

5. Case study

This section takes the generation of the TSSs of the output shaft of the gear reducer in Fig. 8 and the tolerance zones of these TSSs as an example to illustrate how the designed generation method of TSSs and corresponding tolerance zones works. According to the designed method, the TSSs of the output shaft and their tolerance zones are generated by the following steps:

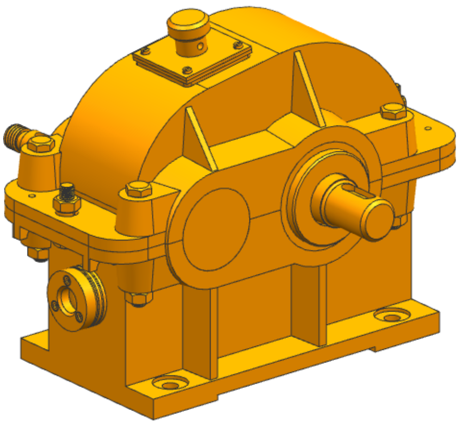


Fig. 8 A gear reducer designed in a CAD system

- Extract information from assembly. Using the API of the CAD system, the information of components, mating relations between components, geometric features, mating relations between geometric features, and constraints between geometric features is extracted from the gear reducer which is designed by the CAD system. Since the section mainly explains the process of the generation of the TSSs of the output shaft and their tolerance zones, this step and the remaining steps only concerns the output shaft (component *c* in Fig. 9) and the adjusting ring, sleeve, flat key, gear, bearing, and seal ring (components c_1, c_2, c_3, c_4, c_5, and c_6 in Fig. 9) that have mating relations with it for the sake of simplicity. The extracted information of the mating relations between *c*, c_1, c_2, c_3, c_4, c_5, and c_6, the extracted information of the mating relations between the geometric features of *c*, c_1, c_2, c_3, c_4, c_5, and c_6, and the extracted information of the constraints between the geometric features of *c*, c_1, c_2, c_3, c_4, c_5, and c_6 are shown in Fig. 10, Fig. 11, and Fig. 12, respectively.

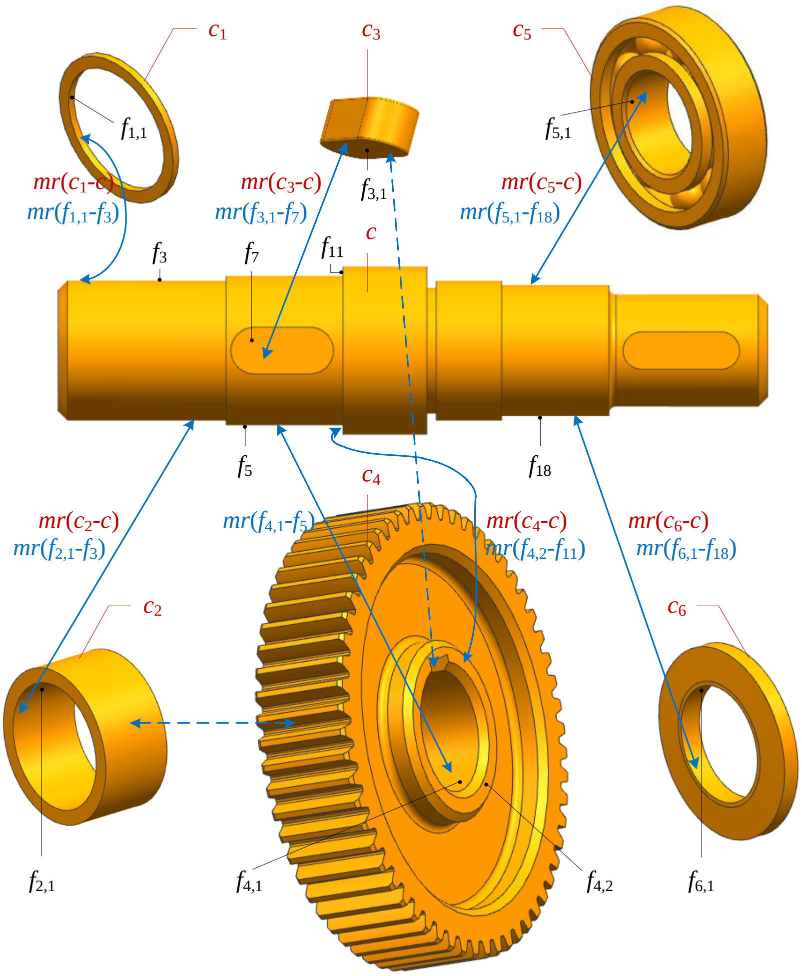
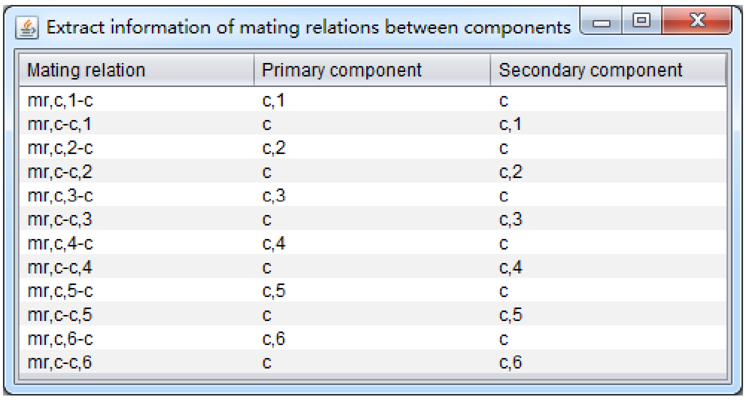


Fig. 9 The output shaft and the components having mating relations with it



Mating relation	Primary component	Secondary component
mr,c,1-c	c,1	c
mr,c-c,1	c	c,1
mr,c,2-c	c,2	c
mr,c-c,2	c	c,2
mr,c,3-c	c,3	c
mr,c-c,3	c	c,3
mr,c,4-c	c,4	c
mr,c-c,4	c	c,4
mr,c,5-c	c,5	c
mr,c-c,5	c	c,5
mr,c,6-c	c,6	c
mr,c-c,6	c	c,6

Fig. 10 The extracted information of the mating relations between c, c_1, c_2, c_3, c_4, c_5, and c_6[†]

[†]Actually, a mating relation between c_2 and c_4 and a mating relation between c_3 and c_4 have been extracted. But they are not displayed in this figure because they have no effect on the TSSs of the output shaft c. For the same reason, the mating relations, constraints, and VGCs between the geometric features of c_2, c_3, and c_4 will not be respectively displayed in Fig. 11, Fig. 12, and Fig. 13.

Extract information of mating relations between features

Mating relation	Primary feature	Secondary feature
mr,f,1,1-f,3	Cylindrical surface f,1,1	Cylindrical surface f,3
mr,f,2,1-f,3	Cylindrical surface f,2,1	Cylindrical surface f,3
mr,f,3,1-f,7	Planar surface f,3,1	Planar surface f,7
mr,f,4,1-f,5	Cylindrical surface f,4,1	Cylindrical surface f,5
mr,f,4,2-f,11	Planar surface f,4,2	Planar surface f,11
mr,f,5,1-f,18	Cylindrical surface f,5,1	Cylindrical surface f,18
mr,f,6,1-f,18	Cylindrical surface f,6,1	Cylindrical surface f,18

Fig. 11 The extracted information of the mating relations between the geometric features of c, c_1, c_2, c_3, c_4, c_5, and c_6

Extract information of constraints between features

Constraint	Referenced feature	Constrained feature	Geometric relation	Variational parameter
c,adf,1,1-f,1,1	Straight line adf,1,1	Cylindrical surface f,1,1	Constraint relation cr,adf,1,1-f,1,1	T(2), R(2)
c,adf,2,1-f,2,1	Straight line adf,2,1	Cylindrical surface f,2,1	Constraint relation cr,adf,2,1-f,2,1	T(2), R(2)
c,adf,3,1-f,3,1	Planar surface adf,3,1	Planar surface f,3,1	Constraint relation cr,adf,3,1-f,3,1	T(1), R(2)
c,adf,4,1-f,4,1	Straight line adf,4,1	Cylindrical surface f,4,1	Constraint relation cr,adf,4,1-f,4,1	T(2), R(2)
c,adf,4,2-f,4,2	Planar surface adf,4,2	Planar surface f,4,2	Constraint relation cr,adf,4,2-f,4,2	T(1), R(2)
c,adf,5,1- f,5,1	Straight line adf,5,1	Cylindrical surface f,5,1	Constraint relation cr,adf,5,1- f,5,1	T(2), R(2)
c,adf,6,1-f,6,1	Straight line adf,6,1	Cylindrical surface f,6,1	Constraint relation cr,adf,6,1-f,6,1	T(2), R(2)
c,adf,3-f,3	Straight line adf,3	Cylindrical surface f,3	Constraint relation cr,adf,3-f,3	T(2), R(2)
c,adf,7-f,7	Planar surface adf,7	Planar surface f,7	Constraint relation cr,adf,7-f,7	T(1), R(2)
c,adf,5-f,5	Straight line adf,5	Cylindrical surface f,5	Constraint relation cr,adf,5-f,5	T(2), R(2)
c,adf,11-f,11	Planar surface adf,11	Planar surface f,11	Constraint relation cr,adf,11-f,11	T(1), R(2)
c,adf,18-f,18	Straight line adf,18	Cylindrical surface f,18	Constraint relation cr,adf,18-f,18	T(2), R(2)
c,adf,1,1-adf,3	Straight line adf,1,1	Straight line adf,3	Coincident relation cr,adf,1,1-adf,3	T(2), R(2)
c,adf,2,1-adf,3	Straight line adf,2,1	Straight line adf,3	Coincident relation cr,adf,2,1-adf,3	T(2), R(2)
c,adf,3,1-adf,7	Planar surface adf,3,1	Planar surface adf,7	Coincident relation cr,adf,3,1-adf,7	T(1)
c,adf,4,1-adf,5	Straight line adf,4,1	Straight line adf,5	Coincident relation cr,adf,4,1-adf,5	T(2), R(2)
c,adf,4,2-adf,11	Planar surface adf,4,2	Planar surface adf,11	Coincident relation cr,adf,4,2-adf,11	T(1)
c,adf,5,1-adf,18	Straight line adf,5,1	Straight line adf,18	Coincident relation cr,adf,5,1-adf,18	T(2), R(2)
c,adf,6,1-adf,18	Straight line adf,6,1	Straight line adf,18	Coincident relation cr,adf,6,1-adf,18	T(2), R(2)
c,f,1,1-f,3	Cylindrical surface f,1,1	Cylindrical surface f,3	Mating relation mr,f,1,1-f,3	T(1), R(1)
c,f,2,1-f,3	Cylindrical surface f,2,1	Cylindrical surface f,3	Mating relation mr,f,2,1-f,3	T(1), R(1)
c,f,3,1-f,7	Planar surface f,3,1	Planar surface f,7	Mating relation mr,f,3,1-f,7	T(1), R(1)
c,f,4,1-f,5	Cylindrical surface f,4,1	Cylindrical surface f,5	Mating relation mr,f,4,1-f,5	T(1), R(1)
c,f,4,2-f,11	Planar surface f,4,2	Planar surface f,11	Mating relation mr,f,4,2-f,11	T(1), R(1)
c,f,5,1-f,18	Cylindrical surface f,5,1	Cylindrical surface f,18	Mating relation mr,f,5,1-f,18	T(1), R(1)
c,f,6,1-f,18	Cylindrical surface f,6,1	Cylindrical surface f,18	Mating relation mr,f,6,1-f,18	T(1), R(1)

Fig. 12 The extracted information of the constraints between the geometric features of c, c_1, c_2, c_3, c_4, c_5, and c_6

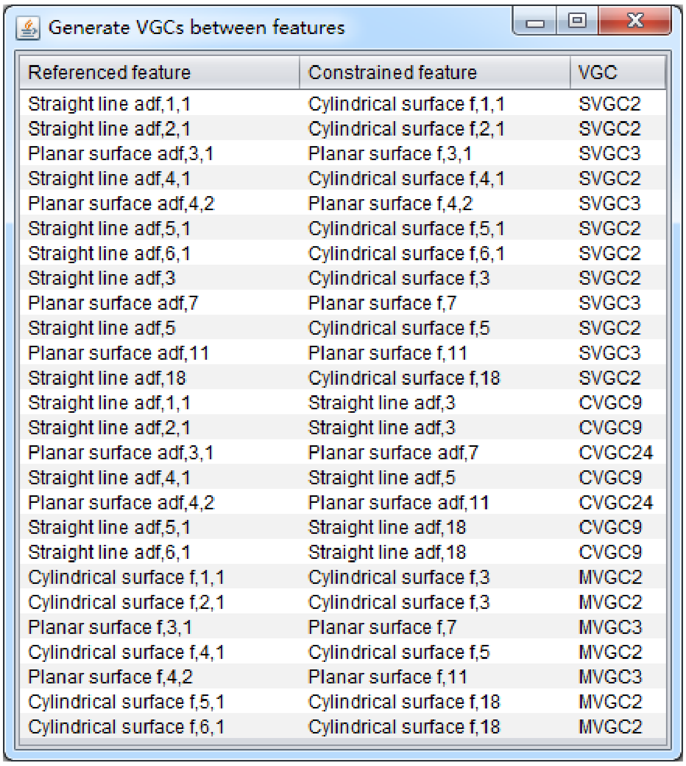
- Construct matrices for mating relations. According to the extracted information in Fig. 10 and Fig. 11, Definition 1, and Definition 2, an adjacency matrix for the mating relations between c, c_1, c_2, c_3, c_4, c_5, and c_6 ($\boldsymbol{M}_{\mathrm{C},7\times 7}$) and an adjacency matrix for the mating relations between the geometric features of c, c_1, c_2, c_3, c_4, c_5, and c_6 ($\boldsymbol{M}_{\mathrm{GF},7\times 5}$) are respectively constructed as follows:

$$\boldsymbol{M}_{\mathrm{C},7\times 7} = \begin{bmatrix} 0 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 1 & 1 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$
\boldsymbol{M}_{\mathbf{GF},7\times5} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}
$$

where the rows of $\boldsymbol{M}_{\mathbf{C},7\times7}$ respectively stands for c, c_1, c_2, c_3, c_4, c_5, and c_6; the columns of $\boldsymbol{M}_{\mathbf{C},7\times7}$ respectively stands for c, c_1, c_2, c_3, c_4, c_5, and c_6; the rows of $\boldsymbol{M}_{\mathbf{GF},7\times5}$ respectively stands for cylindrical surface $f_{1,1}$, cylindrical surface $f_{2,1}$, planar surface $f_{3,1}$, cylindrical surface $f_{4,1}$, planar surface $f_{4,2}$, cylindrical surface $f_{5,1}$, and cylindrical surface $f_{6,1}$; and the columns of $\boldsymbol{M}_{\mathbf{GF},7\times5}$ respectively stands for cylindrical surface f_3, cylindrical surface f_5, planar surface f_7, planar surface f_{11}, and cylindrical surface f_{18}.

- Generate VGCs. On the basis of the extracted information in Fig. 12 and the definition of each type of VGC in [27], the VGCs between the geometric features of c, c_1, c_2, c_3, c_4, c_5, and c_6 are generated and shown in Fig. 13.



Referenced feature	Constrained feature	VGC
Straight line adf,1,1	Cylindrical surface f,1,1	SVGC2
Straight line adf,2,1	Cylindrical surface f,2,1	SVGC2
Planar surface adf,3,1	Planar surface f,3,1	SVGC3
Straight line adf,4,1	Cylindrical surface f,4,1	SVGC2
Planar surface adf,4,2	Planar surface f,4,2	SVGC3
Straight line adf,5,1	Cylindrical surface f,5,1	SVGC2
Straight line adf,6,1	Cylindrical surface f,6,1	SVGC2
Straight line adf,3	Cylindrical surface f,3	SVGC2
Planar surface adf,7	Planar surface f,7	SVGC3
Straight line adf,5	Cylindrical surface f,5	SVGC2
Planar surface adf,11	Planar surface f,11	SVGC3
Straight line adf,18	Cylindrical surface f,18	SVGC2
Straight line adf,1,1	Straight line adf,3	CVGC9
Straight line adf,2,1	Straight line adf,3	CVGC9
Planar surface adf,3,1	Planar surface adf,7	CVGC24
Straight line adf,4,1	Straight line adf,5	CVGC9
Planar surface adf,4,2	Planar surface adf,11	CVGC24
Straight line adf,5,1	Straight line adf,18	CVGC9
Straight line adf,6,1	Straight line adf,18	CVGC9
Cylindrical surface f,1,1	Cylindrical surface f,3	MVGC2
Cylindrical surface f,2,1	Cylindrical surface f,3	MVGC2
Planar surface f,3,1	Planar surface f,7	MVGC3
Cylindrical surface f,4,1	Cylindrical surface f,5	MVGC2
Planar surface f,4,2	Planar surface f,11	MVGC3
Cylindrical surface f,5,1	Cylindrical surface f,18	MVGC2
Cylindrical surface f,6,1	Cylindrical surface f,18	MVGC2

Fig. 13 The generated VGCs between the geometric features of c, c_1, c_2, c_3, c_4, c_5, and c_6

- Construct matrices for VGCs. According to the generated VGCs in Fig. 13 and Definition 3, three adjacency matrices for the VGCs between geometric features ($\boldsymbol{M}_{\mathbf{SVGC},12\times12}$, $\boldsymbol{M}_{\mathbf{CVGC},7\times5}$, and $\boldsymbol{M}_{\mathbf{MVGC},7\times5}$) are constructed as follows:

$$\boldsymbol{M}_{\mathbf{SVGC},12\times12}=\begin{bmatrix} s_2 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 \\ s_0 & s_2 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 \\ s_0 & s_0 & s_3 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 \\ s_0 & s_0 & s_0 & s_2 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 \\ s_0 & s_0 & s_0 & s_0 & s_3 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 \\ s_0 & s_0 & s_0 & s_0 & s_0 & s_2 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 \\ s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_2 & s_0 & s_0 & s_0 & s_0 & s_0 \\ s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_2 & s_0 & s_0 & s_0 & s_0 \\ s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_2 & s_0 & s_0 & s_0 \\ s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_3 & s_0 & s_0 \\ s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_3 & s_0 \\ s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_0 & s_2 \end{bmatrix}$$

$$\boldsymbol{M}_{\mathbf{CVGC},7\times5}=\begin{bmatrix} c_9 & c_0 & c_0 & c_0 & c_0 \\ c_9 & c_0 & c_0 & c_0 & c_0 \\ c_0 & c_0 & c_{24} & c_0 & c_0 \\ c_0 & c_9 & c_0 & c_0 & c_0 \\ c_0 & c_0 & c_0 & c_{24} & c_0 \\ c_0 & c_0 & c_0 & c_0 & c_9 \\ c_0 & c_0 & c_0 & c_0 & c_9 \end{bmatrix}$$

$$\boldsymbol{M}_{\mathbf{MVGC},7\times5}=\begin{bmatrix} m_2 & m_0 & m_0 & m_0 & m_0 \\ m_2 & m_0 & m_0 & m_0 & m_0 \\ m_0 & m_0 & m_3 & m_0 & m_0 \\ m_0 & m_2 & m_0 & m_0 & m_0 \\ m_0 & m_0 & m_0 & m_3 & m_0 \\ m_0 & m_0 & c_0 & m_0 & m_2 \\ m_0 & m_0 & c_0 & m_0 & m_2 \end{bmatrix}$$

where the rows of $\boldsymbol{M}_{\mathbf{SVGC},12\times12}$ respectively stands for straight line $adf_{1,1}$, straight line $adf_{2,1}$, plane $adf_{3,1}$, straight line $adf_{4,1}$, plane $adf_{4,2}$, straight line $adf_{5,1}$, straight line $adf_{6,1}$, straight line adf_3, plane adf_7, straight line adf_5, plane adf_{11}, and straight line adf_{18}; the columns of $\boldsymbol{M}_{\mathbf{SVGC},12\times12}$ respectively stands for cylindrical surface $f_{1,1}$, cylindrical surface $f_{2,1}$, planar surface $f_{3,1}$, cylindrical surface $f_{4,1}$, planar surface $f_{4,2}$, cylindrical surface $f_{5,1}$, cylindrical surface $f_{6,1}$, cylindrical surface f_3, cylindrical surface f_5, planar surface f_7, planar surface f_{11}, and cylindrical surface f_{18}; the rows of $\boldsymbol{M}_{\mathbf{CVGC},7\times5}$ respectively denotes straight line $adf_{1,1}$, straight line $adf_{2,1}$, plane $adf_{3,1}$, straight line $adf_{4,1}$, plane $adf_{4,2}$, straight line $adf_{5,1}$, and straight line $adf_{6,1}$; the columns of $\boldsymbol{M}_{\mathbf{CVGC},7\times5}$ respectively denotes straight line adf_3, straight line adf_5, plane adf_7, plane adf_{11}, and straight line adf_{18}; the rows of $\boldsymbol{M}_{\mathbf{MVGC},7\times5}$ respectively stands for cylindrical surface $f_{1,1}$, cylindrical surface $f_{2,1}$, planar surface $f_{3,1}$, cylindrical surface $f_{4,1}$, planar surface $f_{4,2}$, cylindrical surface $f_{5,1}$, and cylindrical surface $f_{6,1}$; and the columns of $\boldsymbol{M}_{\mathbf{MVGC},7\times5}$ respectively stands for cylindrical surface f_3, cylindrical surface f_5, planar surface f_7, planar surface f_{11}, and cylindrical surface f_{18}.

- Select datum. Since c is a shaft component, the common axis of the cylindrical surfaces f_3 and f_{20} (i.e. the axis A–B in Fig. 14) can be selected as the datum of this component.

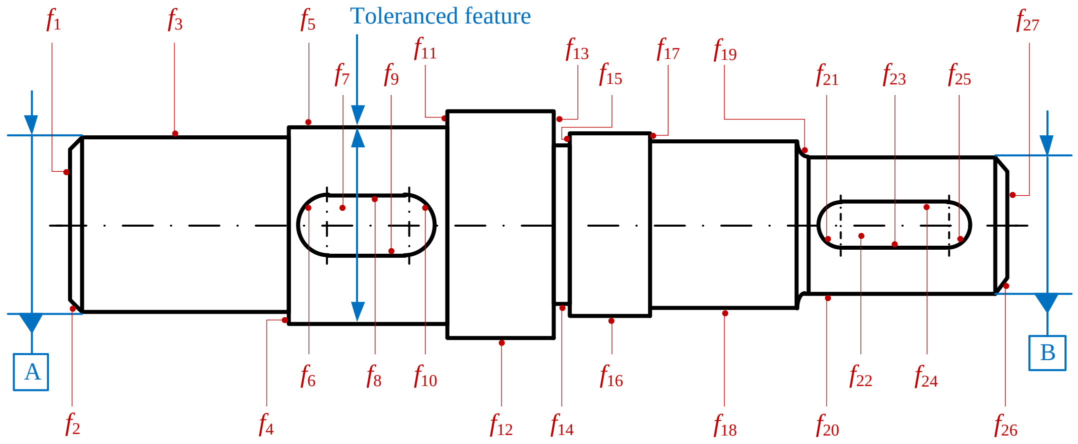


Fig. 14 Notations of all geometric features of the output shaft

- Identify toleranced features. According to the extracted information of the mating relations between the geometric features in Fig. 11, the mating features of c are the cylindrical surface f_3, cylindrical surface f_5, planar surface f_7, planar surface f_{11}, cylindrical surface f_{18}, and cylindrical surface f_{20} (f_{20} will have mating relation with a geometric feature of a output device). Because using default tolerance for f_7 can make it meet operating requirement and the data feature has been selected as the common axis of f_3 and f_{20}, the toleranced features are identified as f_3, f_5, f_{11}, f_{18}, and f_{20}. In addition, the feature of size of f_5 (i.e. its axis A(f_5)) has an important effect on the function of c. Thus A(f_5) is selected as a toleranced feature.
- Generate and select tolerance types. The possible tolerance types for each toleranced feature are generated and shown in Fig. 15 according to the constructed adjacency matrices $\boldsymbol{M}_{\mathbf{SVGC},12\times12}$, $\boldsymbol{M}_{\mathbf{CVGC},7\times5}$, and $\boldsymbol{M}_{\mathbf{MVGC},7\times5}$ and the mapping relations from each type of VGC to tolerance types in [27]. Then according to the actual function of each toleranced feature, the following selections are made: for A(f_5), straightness and coaxiality are selected; for f_5, circular run-out is selected; for f_{11}, total run-out is selected; for f_{20}, circular run-out is selected; for f_{18}, cylindricity and circular run-out are selected; for f_3, cylindricity and circular run-out are selected.

Component	Toleranced feature	Tolerance types
c	Axis of cylindrical surface f,5	Straightness; Roundness; Cylindricity; Coaxiality; Circular run-out; Total run-out
c	Cylindrical surface f,5	Straightness; Roundness; Cylindricity; Coaxiality; Circular run-out; Total run-out
c	Planar surface f,11	Straightness; Flatness; Position; Symmetry; Circular run-out; Total run-out
c	Cylindrical surface f,20	Straightness; Roundness; Cylindricity; Coaxiality; Circular run-out; Total run-out
c	Cylindrical surface f,18	Straightness; Roundness; Cylindricity; Coaxiality; Circular run-out; Total run-out
c	Cylindrical surface f,3	Straightness; Roundness; Cylindricity; Coaxiality; Circular run-out; Total run-out

Fig. 15 The generated possible tolerance types for the tolerance features of *c*

- Consider applying tolerance principles. Based on the specified tolerance types for A(f_5) and the actual function of this toleranced feature, a general tolerance principle maximum material requirement is applied to the specified straightness.
- Generate TSSs. On the basis of the selected datum for *c*, identified toleranced features for *c*, specified tolerance types for toleranced features, and applied tolerance principle for tolerance type, the TSSs for *c* are generated and shown in Fig. 16. The graphic representation of these TSSs is depicted in Fig. 17.

Component	TSS	Toleranced feature	Type	Value	GTP	Primary datum	PDTP	Secondary datum	SDTP	Tertiary datum	TDTP
c	tss,1	Axis of cylindrical surface f,5	Straightness	phi,t,1	MMR	Null	Null	Null	Null	Null	Null
c	tss,2	Axis of cylindrical surface f,5	Coaxiality	phi,t,2	Null	Datum line A-B	Null	Null	Null	Null	Null
c	tss,3	Cylindrical surface f,5	Circular run-out	t,3	Null	Datum line A-B	Null	Null	Null	Null	Null
c	tss,4	Planar surface f,11	Total run-out	t,4	Null	Datum line A-B	Null	Null	Null	Null	Null
c	tss,5	Cylindrical surface f,20	Circular run-out	t,5	Null	Datum line A-B	Null	Null	Null	Null	Null
c	tss,6	Cylindrical surface f,18	Cylindricity	t,6	Null	Null	Null	Null	Null	Null	Null
c	tss,7	Cylindrical surface f,18	Circular run-out	t,7	Null	Datum line A-B	Null	Null	Null	Null	Null
c	tss,8	Cylindrical surface f,3	Cylindricity	t,8	Null	Null	Null	Null	Null	Null	Null
c	tss,9	Cylindrical surface f,3	Circular run-out	t,9	Null	Datum line A-B	Null	Null	Null	Null	Null

Fig. 16 The generated TSSs for *c*. GTP denotes general tolerance principle. PDTP denotes primary datum tolerance principle. SDTP denotes secondary datum tolerance principle. TDTP denotes tertiary datum tolerance principle

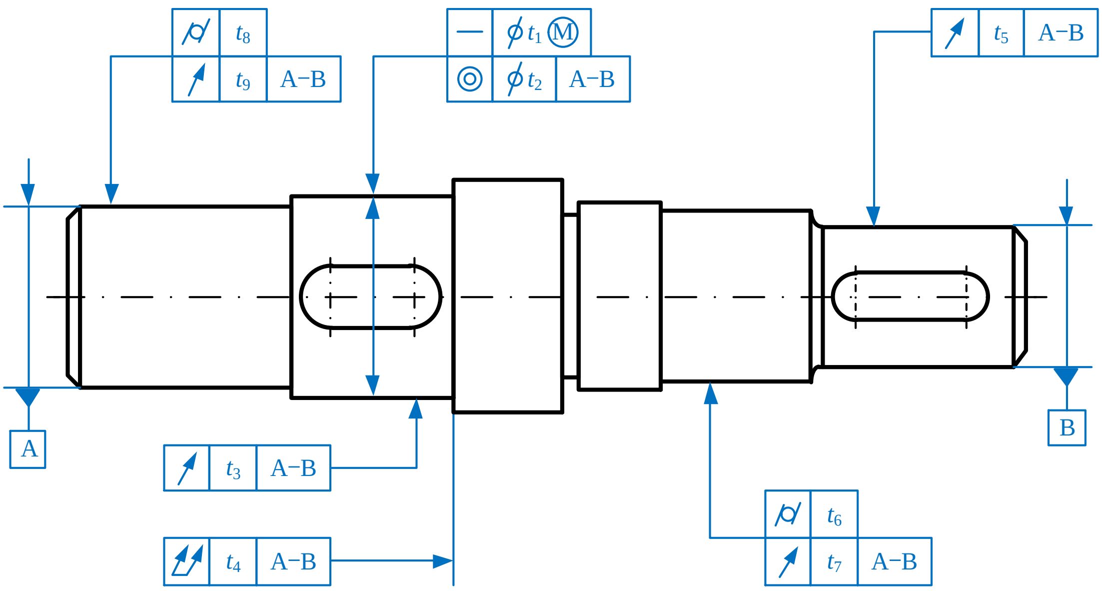


Fig. 17 Graphic representation of the generated TSSs for *c*

- Construct matrix for TSSs. According to the generated TSSs in Fig. 16 and Definition 4, an adjacency matrix for the TSSs of *c* ($\boldsymbol{M}_{\mathrm{TSS},9\times10}$) is constructed as follow:

$$\boldsymbol{M}_{\mathrm{TSS},9\times10} = \begin{bmatrix} \mathrm{A}(f_5) & \mathrm{tt}_3 & \phi t_1 & \textcircled{M} & 0 & 0 & 0 & 0 & 0 & 0 \\ \mathrm{A}(f_5) & \mathrm{tt}_{14} & \phi t_2 & 0 & \mathrm{A{-}B} & 0 & 0 & 0 & 0 & 0 \\ f_5 & \mathrm{tt}_{16} & t_3 & 0 & \mathrm{A{-}B} & 0 & 0 & 0 & 0 & 0 \\ f_{11} & \mathrm{tt}_{17} & t_4 & 0 & \mathrm{A{-}B} & 0 & 0 & 0 & 0 & 0 \\ f_{20} & \mathrm{tt}_{16} & t_5 & 0 & \mathrm{A{-}B} & 0 & 0 & 0 & 0 & 0 \\ f_{18} & \mathrm{tt}_6 & t_6 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ f_{18} & \mathrm{tt}_{16} & t_7 & 0 & \mathrm{A{-}B} & 0 & 0 & 0 & 0 & 0 \\ f_3 & \mathrm{tt}_6 & t_8 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ f_3 & \mathrm{tt}_{16} & t_9 & 0 & \mathrm{A{-}B} & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

- Generate tolerance zones. According to the constructed adjacency matrix $\boldsymbol{M}_{\mathrm{TSS},9\times10}$, the concrete tolerance types in the generated TSSs of *c* are respectively determined as CTT3, CTT 29, CTT33, CTT38, CTT33, CTT6, CTT33, CTT6, and CTT33. Then the tolerance zone forms of the generated TSSs can be respectively obtained on the basis of the correspondences between concrete tolerance types and tolerance zone forms in Table 2. According to the obtained forms and the orientation and location of the generated TSSs in $\boldsymbol{M}_{\mathrm{TSS},9\times10}$, the tolerance zones of the generated TSSs are generated and shown in Fig. 18. The graphic representation of these tolerance zones is depicted in Fig. 19.

Generate tolerance zones for tolerance specification schemes

TSS	TZ	Form	Size	Orientation	Location
tss,1	tz,1	Cylinder	phi,t,1	Null	Null
tss,2	tz,2	Cylinder	phi,t,2	Axis is coaxial with datum line A-B	Axis is coaxial with datum line A-B
tss,3	tz,3	Two concentric circles	t,3	Perpendicular to datum line A-B	Centers are on datum line A-B
tss,4	tz,4	Two parallel planes	t,4	Perpendicular to datum line A-B	Perpendicular to datum line A-B
tss,5	tz,5	Two concentric circles	t,5	Perpendicular to datum line A-B	Centers are on datum line A-B
tss,6	tz,6	Two coaxial cylinders	t,6	Null	Null
tss,7	tz,7	Two concentric circles	t,7	Perpendicular to datum line A-B	Centers are on datum line A-B
tss,8	tz,8	Two coaxial cylinders	t,8	Null	Null
tss,9	tz,9	Two concentric circles	t,9	Perpendicular to datum line A-B	Centers are on datum line A-B

Fig. 18 The generated tolerance zones of the generated TSSs for *c*. TZ denotes tolerance zone

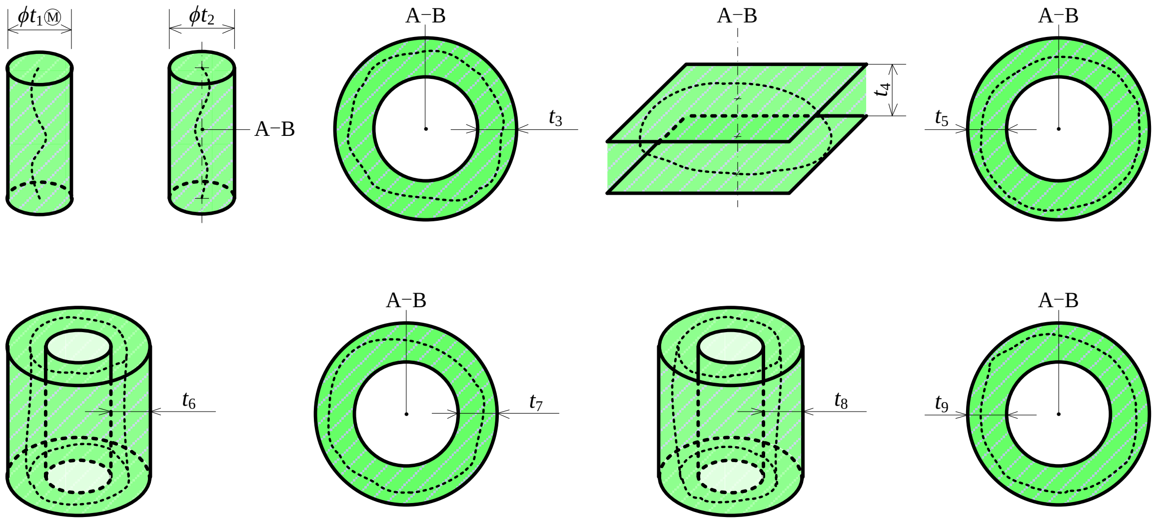


Fig. 19 Graphic representation of the generated tolerance zones of the generated TSSs for $c^{‡}$

- Construct matrix for tolerance zones. According to the generated tolerance zones in Fig. 18 and Definition 5, an adjacency matrix for these tolerance zones ($\boldsymbol{M}_{\mathbf{TZ},9\times4}$) is constructed as follow:

$$\boldsymbol{M}_{\mathbf{TZ},9\times4}=\begin{bmatrix} \mathrm{c} & \phi t_1 & 0 & 0 \\ \mathrm{c} & \phi t_2 & \mathrm{O}_2 & \mathrm{L}_2 \\ \mathrm{e} & t_3 & \mathrm{O}_3 & \mathrm{L}_3 \\ \mathrm{h} & t_4 & \mathrm{O}_4 & \mathrm{L}_4 \\ \mathrm{e} & t_5 & \mathrm{O}_5 & \mathrm{L}_5 \\ \mathrm{d} & t_6 & 0 & 0 \\ \mathrm{e} & t_7 & \mathrm{O}_7 & \mathrm{L}_7 \\ \mathrm{d} & t_8 & 0 & 0 \\ \mathrm{e} & t_9 & \mathrm{O}_9 & \mathrm{L}_9 \end{bmatrix}$$

where O_2 denotes the axis of the tolerance zone is coaxial with A–B; O_3 denotes the tolerance zone is perpendicular to A–B; O_4 denotes the tolerance zone is perpendicular to A–B; O_5 denotes the tolerance zone is perpendicular to A–B; O_7 denotes the tolerance zone is perpendicular to A–B; O_9 denotes the tolerance zone is perpendicular to A–B; L_2 stands for the axis of the tolerance zone is coaxial with A–B; L_3 stands for the centers of the tolerance zone are on A–B; L_4 stands for the tolerance zone is perpendicular to A–B; L_5 stands for the centers of the tolerance zone are on A–B; L_7 stands for the centers of the tolerance zone are on A–B; and L_9 stands for the centers of the tolerance zone are on A–B.

6. Comparison and discussion

In general, a quantitative comparison between different tolerance representation models is difficult to

[‡]The nine tolerance zones in this figure are respectively the generated tolerance zones of the nine TSSs in Fig. 17. In fact, the generated tolerance zones of each pair of TSSs assigned on f_5, f_{18}, and f_3 are three complicated tolerance zones, whose construction method has been specifically studied in [4]. For the details regarding the construction of these tolerance zones, please refer to [4].

be made since it is difficult to quantify the performance of a tolerance representation model. For this reason, a comparison between different tolerance representation models can only be made in a qualitative way. The review of tolerance represenatation models in [2] presented to make such a comparison from the following twelve aspects:

Aspect 1: Is it directly computer-readable?

Aspect 2: Is it directly computer-interpretable?

Aspect 3: Has it unambiguously represented information syntax?

Aspect 4: Has it explicitly represented information semantics?

Aspect 5: Has it included a systematic representation of product geometry?

Aspect 6: Has it included a systematic representation of design requirements?

Aspect 7: Has it been applied in industry?

Aspect 8: Does it adopt a universal tool to represent information?

Aspect 9: Does it provide an interface for information exchange?

Aspect 10: Is it easy to be reused and extended?

Aspect 11: Does it have intrinsic conformance checking, query, and reasoning capabilities?

Aspect 12: Does it have self-learning and automatic update capabilities?

On the basis of these aspects, a qualitative comparison between the presented tolerance representation model in this paper and the existing eight categories of tolerance representation models reviewed in Section 2 is carried out in this section. The results of this qualitative comparison are shown in Table 3. It can be seen from Table 3 that eight different categories of tolerance representation models are involved in the qualitative comparison. Since the mathematical definition based model [5–12], TTRS model [21], graph-based model [26–29], modeling language model [30–35], category theory model [37], polychromatic set model [41, 42], and ontology-based model [43–47] are also involved in the qualitative comparison of the review in [2] and the review has presented a comprehensive analysis of the results of the qualitative comparison between these models (i.e. the columns from M1 to M7 in Table 3), such an analysis will not be provided here. For the details regarding the analysis, please refer to [2]. As for the presented model, its result is listed in the column M8 in Table 3. This result is analyzed as follows:

- Aspect 1: The physical manifestation of the presented model is a set of adjacency matrices, whose computer readability is achieved with the assist of the Java programming language. Thus, the presented model is not directly computer-readable.
- Aspect 2: The presented model is not directly computer-interpretable because adjacency matrices are not based on formal semantics.

- Aspect 3: It is no doubt that the presented model has unambiguously represented the syntax of tolerance information, because such capability is the most fundamental capability of a tolerance representation model.
- Aspect 4: The semantics of tolerance information are implicitly represented but not explicitly represented. Explicitly representing the semantics of information requires specific semantic representation languages (e.g. OWL). Adjacency matrix does not have such capability.
- Aspect 5: All of the things represented by adjacency matrices (e.g. components and the mating relations between them, geometric features and the mating relations between them) belong to product geometry. The presented model has provided a systematic representation of these things.
- Aspect 6: Design requirements mainly include functional and assembly requirements. The presented model has only included a small part of them. Systematically representing design requirements for tolerancing remains a challenging issue.
- Aspect 7: The presented model has not yet been applied in industry.
- Aspect 8: The representation tool adopted in the presented model is adjacency matrix, which is not a universal representation tool as defined in [2].
- Aspect 9: The presented model cannot be directly exchanged among different systems because it is not directly computer-readable.
- Aspect 10: Generally, a model is easy to be reused and extended by others if it adopts a universal representation tool to represent information [2]. Since adjacency matrix is not a universal representation tool as defined in [2], the presented model is not easy to be reused and extended.
- Aspect 11: Because adjacency matrix does not have intrinsic conformance checking, query, and reasoning capabilities, the presented model also does not have such capabilities.
- Aspect 12: The presented model does not have self-learning and automatic update capabilities since self-learning and automatic update mechanisms have not yet been introduced into it.

Table 3 A qualitative comparison between the presented tolerance representation model in this paper and the existing eight categories of tolerance representation models reviewed in Section 2. M1 stands for mathematical definition based model [5–12]. M2 stands for TTRS model [21]. M3 stands for graph-based model [26–29]. M4 stands for modeling language model [30–35]. M5 stands for category theory model [37]. M6 stands for polychromatic set model [41, 42]. M7 stands for ontology-based model [43–47]. M8 stands for the presented model (Please note that the presented model can be seen as an extended model of the adjacency matrix model [3]. Thus there is no separate adjacency matrix model in this table). In addition to the M8 column, the remaining columns are quoted from the review of tolerance represenatation models in [2].

Comparison aspect	M1	M2	M3	M4	M5	M6	M7	M8

Aspect 1	NO	NO	NO	YES	NO	NO	YES	NO
Aspect 2	NO	NO	NO	NO	NO	NO	YES	NO
Aspect 3	YES	YES	YES	YES	YES	YES	YES	YES
Aspect 4	DK	DK	DK	DK	DK	DK	YES	NO
Aspect 5	YES	YES	YES	YES	YES	YES	YES	YES
Aspect 6	NO	NO	NO	NO	NO	NO	NO	NO
Aspect 7	YES	YES	DK	DK	DK	DK	DK	NO
Aspect 8	NO	NO	NO	YES	NO	NO	YES	NO
Aspect 9	NO	NO	NO	YES	NO	NO	YES	NO
Aspect 10	DK	DK	DK	YES	DK	DK	YES	NO
Aspect 11	NO	NO	NO	NO	NO	NO	YES	NO
Aspect 12	NO	NO	NO	NO	NO	NO	NO	NO

As can been seen from the analysis above, the presented model only has advantages in Aspect 3 and Aspect 5 according to the comparison benchmark defined in [2]. Nevertheless, it still cannot be simply concluded that the presented model is not as good as some other models (e.g. modeling language model, ontology-based model), because difference models may have advantages and disadvantages in different aspects and the comparison benchmark defined in [2] is not comprehensive enough (It is commonly believed that to define a complete qualitative comparison benchmark is currently impossible). From this point of view, whether a new tolerance representation model is of necessity depends on its specific situation and purpose.

The purpose of the current paper is to present a tolerance representation model for generating TSSs and corresponding tolerance zones to compensate for the deficiencies of the adjacency matrix model in [3]. As stated in the introduction, the model in [3] does not involve the representation of VGCs and resultant tolerance zones and only address the generation of tolerance types. Compared to this model, the presented model not only represents VGCs and resultant tolerance zones, but also takes into account the generation of TSSs and corresponding tolerance zones. Thus at this point of view, the presented model is of theoretical significance. Finally, it should be pointed out that the presented model is only an alternative tolerance representation model and it should not be seen as a replacement of other categories of models.

7. Conclusion

To meet the demand of representing the semantics of tolerance information, a tolerance representation model for generating tolerance specification schemes and corresponding tolerance zones has been proposed in this paper. This model consists of component layer, geometric feature layer, VGC layer, TSS layer, and tolerance zone layer. The knowledge in each of these layers is formalized by one or more adjacency matrices, respectively. Based on the model, a method for generating tolerance specification schemes for component and their resultant tolerance zones is designed. This method shows how to adopt a top-down strategy to implement computer-aided tolerance specification. The paper has also provided a practical example to illustrate the working process of the method.

Future research will aim especially at overcoming one main limitation of the proposed tolerance representation model: The proposed model mainly concerns geometric information and a small amount of design requirements information (including functional requirement information and assembly requirement information). This is not comprehensive enough since geometric information, design requirements information, process information, and material information are required to be synthetically considered in tolerance specification. Hence, it is of significance to extend the model to include such information. In addition, as tolerance specification, tolerance allocation and analysis are also two important parts of tolerancing, it is of necessity to study computer-aided tolerance allocation and analysis on the basis of the extended model.

Acknowledgements

The authors would like to appreciate the insightful comments from the three anonymous reviewers and the regional editor Professor Andrew Y. C. Nee for the improvement of the paper. The authors also would like to acknowledge the financial supports by the National Natural Science Foundation of China (Nos. 51475190 and 51365009), the Hubei Provincial Natural Science Foundation of China (No. 2015CFA109), the Doctoral Dissertation Innovation Foundation of Huazhong University of Science and Technology, and the National Scholarship of China Scholarship Council.

References

[1] Guilford J, Turner J (1993) Representational primitives for geometric tolerancing. Comput Aided Des 25(9):577–586

[2] Qin Y, Qi Q, Lu W, Liu X, Scott PJ, Jiang X (2018) A review of representation models of tolerance information. Int J Adv Manuf Tech 95(5–8):2193–2206

[3] Qin Y, Zhong Y, Huang M, Liu F (2014) An assembly tolerance representation model based on spatial relations for generating assembly tolerance types. Proc IMechE Part C: J Mech Eng Sci 228(6):1005–1020

[4] Liu Y, Gao S, Cao Y (2009) An efficient approach to interpreting rigorous tolerance semantics for complicated tolerance specification. IEEE Trans Autom Sci Eng 6(4):670–684

[5] Franciosa P, Gerbino S, Patalano S (2010) Variational modeling and assembly constraints in tolerance analysis of rigid part assemblies: Planar and cylindrical features. Int J Adv Manuf Technol 49 (1–4):239–251

[6] Pasupathy TMK, Morse EP, Wilhelm RG (2003) A survey of mathematical methods for the construction of geometric tolerance zones. ASME Trans J Comput Inf Sci Eng 3(1):64–75

[7] Shah JJ, Yan Y, Zhang BC (1998) Dimension and tolerance modeling and transformations in feature based design and manufacturing. J Intell Manuf 9(5):475–488

[8] Roy U, Li B (1998) Representation and interpretation of geometric tolerances for polyhedral objects I: form tolerances. Comput Aided Des 30(2):151–161

[9] Roy U, Li B (1999) Representation and interpretation of geometric tolerances for polyhedral objects II: size, orientation and position tolerances. Comput Aided Des 31(4):273–285

[10] Davidson JK, Mujezinovic A, Shah JJ (2002) A new mathematical model for geometric tolerances as applied to round faces. ASME Trans J Mech Des 124(4):609–622

[11] Mujezinovic A, Davidson JK, Shah JJ (2004) A new mathematical model for geometric tolerances as applied to polygonal faces. ASME Trans J Mech Des 126(3):504–518

[12] Ameta G, Davidson JK, Shah JJ (2007) Tolerance-maps applied to a point-line cluster of features. ASME Trans J Mech Des 129(8):782–792

[13] Wu Y, Shah JJ, Davidson JK (2003) Computer modeling of geometric variations in mechanical parts and assemblies. ASME Trans J Comput Inf Sci Eng 3(1):54–63

[14] Ballu A, Mathieu L (1999) Choice of functional specifications using graphs within the framework of education. Proc. 6th CIRP Int. Conf. Comput. Aided Tolerancing. Springer Nertherlands, Houten, p 197–206

[15] Wang H, Roy U, Sudarsan R, Sriram RD, Lyons KW (2003) Functional tolerancing of a gearbox. Proc. North Am. Manuf. Res. Conf. National Institute of Standards and Technology, Gaithersburg, p 1–8

[16] Armillotta A (2013) A method for computer-aided specification of geometric tolerances. Comput Aided Des 45(12):1604–1616

[17] Haghighi P, Mohan P, Kalish N, Vemulapalli P, Shah JJ, Davidson JK (2015) Toward automatic tolerancing of mechanical assemblies: First-order GD&T schema development and tolerance allocation. ASME Trans J Comput Inf Sci Eng 15(4):041003

[18] Anselmetti B, Mawussi K (2003) Computer aided tolerancing using positioning features. ASME Trans J Comput Inf Sci Eng 3(1):15–21

[19] Anselmetti B (2006) Generation of functional tolerancing based on positioning features. Comput Aided Des 38(8):902–919

[20] Cao Y, Zhang H, Li B, Wu Z, Yang J (2013) Study on functional specification scheme on interface based on positioning features. Proc IMechE Part B: J Eng Manuf 227(5):745–753

[21] Desrochers A, Clement A (1994) A dimensioning and tolerancing assistance model for CAD/CAM systems. Int J Adv Manuf Technol 9(6):352–361

[22] Clement A, Riviere A, Serre P, Valade C (1998) The TTRSs: 13 constraints for dimensioning and tolerancing. Proc. 5th CIRP Int. Conf. Comput. Aided Tolerancing. Chapman and Hall, London, p 122–131

[23] Srinivasan V (1999) A geometrical product specification language based on a classification of symmetry group. Comput Aided Des 31(11):659–668

[24] Desrochers A, Maranzana R (1995) Constrained dimensioning and tolerancing assistance for mechanisms. Proc 4th CIRP Int. Sem. Comput. Aided Tolerancing. Springer Netherlands, Houten, p 17–30

[25] Cao Y, Zhao Q, Liu T, Ren L,Yang J (2018) The Strategy of Datum Reference Frame Selection Based on Statistical Learning. ASME Trans J Comput Inf Sci Eng. doi:10.1115/1.4039380

[26] Tsai JC, Cutkosky MR (1997) Representation and reasoning of geometric tolerances in design. Artif Intellect Eng Des Anal Manuf 11(4):325–341

[27] Hu J, Xiong G, Wu Z (2004) A variational geometric constraints network for a tolerance types specification. Int J Adv Manuf Technol 24(3–4):214–222

[28] Franciosa P, Patalano S, Riviere A (2010) 3D tolerance specification: an approach for the analysis of the global consistency based on graphs. Int J Interact Des Manuf 4(1):1–10

[29] Franciosa P, Gerbino S, Lanzotti A, Patalano S (2013) Automatic evaluation of variational parameters for tolerance analysis of rigid parts based on graphs. Int J Interact Design Manuf 7(4):239–248

[30] ISO 10303-242 (2014) Industrial automation systems and integration—Product data representation and exchange—Part 242: Application protocol: Managed model-based 3D engineering. International Organization for Standardization, Geneva

[31] Rachuri S, Han YH, Feng SC, Roy U, Wang F, Sriram RD, Lyons KW (2004) Object-oriented representation of electro-mechanical assemblies using UML. National Institute of Standards and Technology, Gaithersburg

[32] Zhao X, Pasupathy TK, Wilhelm RG (2006) Modeling and representation of geometric tolerances information in integrated measurement processes. Comput Ind 57(4):319–330

[33] Dantan JY, Ballu A, Mathieu L (2008) Geometrical product specifications—Model for product life cycle. Comput Aided Des 40(4):493–501

[34] Ballu A, Dantan JY, Mathieu L (2010) Language of Tolerancing: GeoSpelling. Villeneuve F, Mathieu L, editors. Geometric Tolerancing of Products. John Wiley & Sons, Hoboken, p 23–53

[35] Ballu A, Mathieu L, Dantan JY (2015) Formal language for GeoSpelling. ASME Trans J Comput Inf

Sci Eng 15(2):021002

[36] Wang Y, Scott PJ, Jiang X (2005) The structure of surface texture knowledge. J Phys: Conf Ser 13(1):1–4

[37] Lu W, Jiang X, Liu X, Qi Q, Scott PJ (2010) Modeling the integration between specifications and verification for cylindricity based on category theory. Meas Sci Technol 21(11):115107

[38] Xu Y, Xu Z, Jiang X, Scott P (2011) Developing a knowledge based system for complex geometrical product specification (GPS) data manipulation. Knowl Based Syst 24(1):10–22

[39] Qi Q, Jiang X, Scott PJ (2012) Knowledge modeling for specifications and verification in areal surface texture. Precis Eng 36(2):322–333

[40] Qi Q, Scott PJ, Jiang X, Lu W (2014) Design and implementation of an integrated surface texture information system for design, manufacture and measurement. Comput Aided Des 57(12):41–53

[41] Zhang Y, Li Z, Xu L, Wang J (2011) A new method for automatic synthesis of tolerances for complex assemblies based on polychromatic sets. Enterp Inf Syst 5(3):337–358

[42] Zhang Y, Li Z, Gao J, Hong J (2011) New reasoning algorithm for assembly tolerance specifications and corresponding tolerance zone types. Comput Aided Des 43(12):1606–1628

[43] Fiorentini X, Gambino I, Liang VC, Foufou S, Rachuri R, Mani M, Bock C (2007) An ontology for assembly representation. National Institute of Standards and Technology, Gaithersburg

[44] Lu W, Qin Y, Liu X, Huang M, Zhou L, Jiang X (2015) Enriching the semantics of variational geometric constraint data with ontology. Comput-Aided Des 63(6):72–85

[45] Zhong Y, Qin Y, Huang M, Lu W, Chang L (2014) Constructing a meta-model for assembly tolerance types with a description logic based approach. Comput-Aided Des 48(3):1–16

[46] Qin Y, Lu W, Qi Q, Liu X, Zhou L, Li T (2015) Ontology-based semantic interpretation of cylindricity specification in the next-generation GPS. Proc. 13th CIRP Int. Conf. Comput. Aided Tolerancing. Procedia CIRP 27: 124–130

[47] Qin Y, Lu W, Qi Q, Li T, Huang M, Scott PJ, Jiang X (2017) Explicitly representing the semantics of composite positional tolerance for patterns of holes. Int J Adv Manuf Tech 90(5):2121–2137

[48] Zhong Y, Qin Y, Huang M, Lu W, Gao W, Du Y (2013) Automatically generating assembly tolerance types with an ontology-based approach. Comput Aided Des 45(11):1253–1275

[49] Qin Y, Lu W, Liu X, Huang M, Zhou L, Jiang X (2015) Description logic-based automatic generation of geometric tolerance zones. Int J Adv Manuf Tech 79(5):1221–1237

[50] Qin Y, Lu W, Qi Q, Liu X, Huang M, Scott PJ, Jiang X (2017) Towards an ontology-supported case-based reasoning approach for computer-aided tolerance specification. Knowl Based Syst 141(2):

129–147

[51] Qin Y, Lu W, Qi Q, Liu X, Zhong Y, Scott PJ, Jiang X (2017) Status, comparison, and issues of computer-aided design model data exchange methods based on standardized neutral files and web ontology language file. ASME Trans J Comput Inf Sci Eng 17(1):010801

[52] Ameta G, Singh G, Davidson JK, Shah JJ (2017) Application of T-Maps for Composite Position Tolerance for Patterns of Features. ASME 2017 Int. Des. Eng. Tech. Conf. Comput. Inf. Eng. Conf. American Society of Mechanical Engineers, New York, p V001T02A014-V001T02A014