COMPUTER-AIDED DESIGN & APPLICATIONS, 2017
VOL. 14, NO. 6, 720–733
https://doi.org/10.1080/16864360.2017.1287674

Taylor & Francis
Taylor & Francis Group

A contact relation analysis approach to assembly sequence planning for assembly models

Songqiao Tao [a] and Min Hu [b]

[a]Wuhan Technical College of Communications, China; [b]Huazhong University of Science and Technology, China

ABSTRACT
A reasonable assembly planning can obviously reduce manufacturing time and costs. From the perspective of geometric reasoning in reverse, a contact relation analysis approach to ASP for assembly models is proposed in this paper. The main characteristic of the presented method has an ability to choose non-orthogonal assembling directions automatically. Firstly, contact relations between parts of an assembly model are automatically extracted from its CAD model. According to contact relations, contact vectors are generated. Secondly, the disassembly of a part in an assembly model is checked. Based on extracted contact vectors, instantaneous movable directions for parts are identified by solving inequality constraints. And Gaussian sphere is introduced to express a set of movable directions. Subsequently, currently removable parts are judged by checking the existence of feasible disassembly paths along instantaneous movable directions. Thirdly, different layers of removable parts in an assembly are obtained based on extracted assembly relations and geometric removable judgment. Using the contact state transition analysis of layer graphs of AS and the knowledge of the assembly process, the final assembly sequence of craft viable can be found. Finally, an example has been provided to demonstrate the proposed approach.

KEYWORDS
Assembly sequence planning; geometric reasoning; geometric disassembly; non-orthogonal assembling directions

1. Introduction

Assembly planning is one of the most important tasks in the manufacturing process of mechanical products. Previous research has shown that assembly processes take more than 50% of the total production time and 20% of the total manufacturing cost [21]. Thus, a reasonable assembly planning can obviously reduce manufacturing time and costs. ASP has proved to be an NP-complete problem [17]. So, since recently, ASP has received extensive attention in the academic fields.

The most existing methods for Computer-Aided Process Planning (CAPP) focus on reasoning geometry feasible assembly sequences by analyzing geometry information, topology structure and constraint relationships in an assembly model. To get the optimal assembly sequence, some evaluation mechanisms (such as the lowest cost, highest efficiency, etc.) are introduced to optimize assembly sequences. However, they have some shortcomings in the following three aspects. Firstly, if a complex product has complex parts and a huge number of parts, the time spent on geometry reasoning will be difficult to meet the requirement of practical production. And handling a large amount of feasible ASP is an intractable problem. Secondly, assembly paths of a part is only considered the orthogonal direction for generating ASP, which is unsuitable to a a complex assembly model. Thirdly, there isn't a unified standard of evaluation mechanisms for optimizing ASP.

To resolve the problem that assembly paths of a part is only considered the orthogonal direction for generating ASP, this paper proposed a contact relation analysis approach to ASP from the perspective of geometric reasoning in reverse. The main characteristic of the presented method has an ability to choose non-orthogonal assembling directions automatically. Firstly, contact relations between parts of an assembly model are automatically extracted from its CAD model. A prerequisite is introduced to filter out potential contact surfaces, and using triangular facet tessellation to judge whether surfaces contact. According to contact relations, contact vectors are generated. Secondly, the disassembly of a part in an assembly model is checked. Based on extracted contact vectors, Instantaneous movable directions between parts of an assembly are identified by solving inequality constraints. And Gaussian sphere is introduced to express a set of movable directions.

CONTACT Songqiao Tao ✉ taosongqiao@163.com; Min Hu ✉ 1810069256@qq.com

Subsequently, currently removable parts are judged by checking the existence of feasible disassembly paths along instantaneous movable directions. Finally, different layers of removable parts in an assembly are obtained based on extracted assembly relations and geometric removable judgment. Using the contact state transition analysis of layer graphs of AS and knowledge of the assembly process, the final assembly sequence of craft viable can be found.

The rest of this paper is organized as follows. Firstly, a brief review of related works are presented in Section 2. Then, contact relation analysis for assembly models is introduced in Section 3. In Section 4, the algorithm of generating ASP is given in detail. Following this section, an example has been provided to demonstrate the proposed approach in Section 5. Finally, the paper presents some conclusions and future works in Section 6.

2. Related works

The earlier studies of ASP are appeared in [1–3]. Bourjault [1] was the first to introduce directed graphs and recurring questions for generating assembly sequences (ASs). De Fazio et al. [3] adopted Bourjault's concept to produce a complete set of ASs. In their work, precedence relations between liaisons are firstly created in a product. So, the number of expert questions in De Fazio's approach is obviously reduced. Santocchi et al. [25] used the interference matrix, contact matrix and connection matrix in the Cartesian coordinate system to represent assembly models for analyzing sub-assemblies and ASs. Gu et al. [10] proposed an ordered binary decision graph to describe ASs. To generate and optimize ASs for flexible assembly parts, Ghandiet al. [5] presented assembly stress matrix to represent interference relations between parts and the sizes of compressive stress needed for assembling flexible parts. The graph-based and matrix-based methods mentioned above can represent entire sequences of the solution space in theory. However, the combinatorial nature causes the problem that it is slow and limited for seeking out the optimal solution.

Human-interactive can effectively generate feasible ASP for actual products. Henrioud et al. [11] presented the concept of assembly preferential relations for human-interactive assembly planning. To reduce human input required for generating ASs, Wilson [30] developed an interactive computer-aided assembly planning system. The system allows a user to identify some parts that constrain a subassembly in a query and can automatically use this information to answer future queries. Raghavan et al. [23] introduced the novel human-computer interface of augmented reality to evaluate ASs. The interactive tool considers various sequence alternatives of the manufacturing design process by manipulating virtual and real prototype components. Wang et al. [28] created a manual assembly design system based on augmented reality. The system allows a user to simulate a manual assembly in the absence of auxiliary computer-aided design information. To integrate product design and ASP, Gruhier et al. [9] introduced mereotopology and temporal relationships to represent product relationships. They resolved the problem of information and knowledge inconsistency. The human- interactive methods mentioned above have an advantage that makes good use of human experience. However, the users are easy to feel bored when human-computer interaction is frequent.

In recent years, some knowledge-based engineering methods have been developed to generate ASP automatically. Huang et al. [13] presented a framework of knowledge-based assembly planning for automatic generating assembly plans. Predicate calculus is introduced to represent the knowledge of assembly structure, precedence constraints, and resource constraints in the static knowledge database. Dong et al. [4] used a connection-semantics-based assembly tree to represent geometric information and non-geometric knowledge for generating ASP. Hsu et al. [12] developed a knowledge-based engineering system with a three-stage assembly optimization method to predict a near-optimal AS. They employed graph and a transforming rule, and a robust back-propagation neural network engine to generate feasible ASs. Kashkoushet et al. [16] proposed a knowledge-based mixed-integer programming model to generate AS for a given product based on the existing AS data of its similar products. To improve the efficiency of generating ASP for complicated products, Qual. [22] proposed a methodology-integrated case based-reasoning and constraints-based reasoning. Because dynamic constraints satisfaction and repair algorithm are incorporated, the applicability of the case based-reasoning is extended to a larger class of problems. Son [26] presented a rule-based sequence planning algorithm with fuzzy optimization for generating feasible paths related with apart-bringing in partially dynamic environments. Some knowledge processing functions such as machine reasoning, planning, and decision-making are used in their method. Knowledge based approaches mentioned above take advantage of human experience and knowledge. When the number of parts in an assembly model is very large, it is difficult to analyze geometry and assembly relations.

Furthermore, some artificial intelligence algorithms have been developed to create ASP automatically. Wang et al. [27] proposed a novel ant colony algorithm for generating ASP. To find optimal solutions, least re-orientations are adopted to produce different amount of ants cooperating for different assemblies. Wang et al. [29] proposed a chaotic particle swarm optimization (PSO) method for generating the optimal or near-optimal ASs. They used the velocity operator to replace the variable velocity in the traditional PSO algorithm. Zhang et al. [31] used the immune particle swarm (IPS) algorithm to solve ASP problem. Because IPS algorithm combines the advantage of particle swarm algorithm and immune algorithm, it has high convergence and the ability to overcome the precocious phenomenon.

Because detailed geometry information of a product is easy to extract from its CAD (B-Rep) model, several efforts used CAD models as an input for representing or generating ASP. Laperriére et al. [18] presented a generative assembly process planner from a geometric, stable, and accessible point of view. They used a solid model of the product to be assembled as input and output its feasible ASs. Gottipolu et al. [7, 8] proposed a matrix-based approach for automatic generating ASs. In their method, geometric and mobility constraints are extracted directly from assembly models. Zhang et al. [33] used the connection and contracted matrices to describe the precedence constraint knowledge among components and subassemblies of an automobile body for automatic generating all feasible ASs. Lin et al. [19] presented a contact relation matrix to create ASs for design alternative identification. Ou et al. [20] developed a system that can generate ASs and rank feasible ASs from a CAD model. They used a relationship matrix to process the information retained from a CAD model. Zhang et al. [32] proposed an approach to create interference matrices for generating ASP from a CAD assembly model. For generating and optimizing ASP, more relevant literature can be found in [24–15].

3. Contact relation analysis for assembly models

3.1. Contact types

Fitting, alignment and offset are usually used for expressing mating relations and constraint relations between parts of an assembly model. And these relations form contact relation between parts. In general, contacts between parts of an assembly can be classified into three categories: plane or surface contact (see Figs. 1(a) and (b)), line contact (see Fig. 1(c)) and point contact (see Fig. 1(d)). In Figs. 1(a) and (b), the contact between parts A and B is a plane or surface while the contact between parts A and B in Figs. 1(c) and (d) are respectively a line and a point.

In an assembly model, contact status between parts form kinematic pairs and command the movement of the parts. So, contact status between parts is closely connected with disassembly. Because some rotary parts like gear pairs and screw pairs don't need to be disassembled in contemporary condition, the interference detections only consider translational motion parts for disassembly in this paper.

3.2. Contact status analysis

Contact relation analysis is essentially to judge the surface contact status between parts. To improve the efficiency and accuracy of the contact state analysis, it is necessary to determine contact types. If contact type between parts of an assembly is determined, we may introduce some prerequisites to judge whether planes contact. For plane-plane fitting in Fig. 1(a), outer normal direction and zero distance are the prerequisites for checking whether two planes are fitting. If two planes have opposite outer normal direction, zero distance and common points, and their contact type is determined, we can conclude that two planes are fitting. The prerequisites can eliminate obvious contact status, which effectively improve the analysis efficiency and accuracy. Moreover, plane-plane contacts, plane-surface contacts and surface-surface contacts have in common is that they have common points.

According to above mentioned analysis, we may take three procedures to judge contact status. Firstly, judge contact types between parts of an assembly (see Fig. 1). Secondly, identify judged contact types whether satisfy the prerequisites. Finally, judge whether two surfaces have common points.

To detect whether there are common points between surfaces, a simple method is to select sampling points on two surfaces and to check whether they have common sampling points. Most of commercial CAD modeling

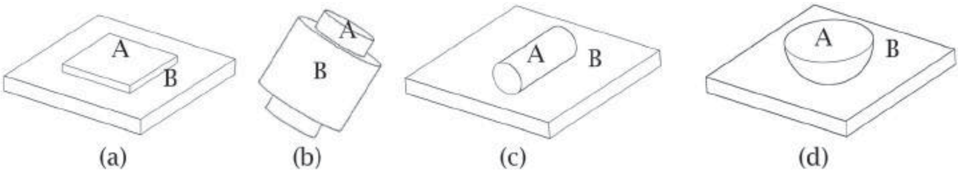


Figure 1. Contact types. (a) Plane contact, (b) Cylinder contact, (c) Line contact, and (d) Point contact.

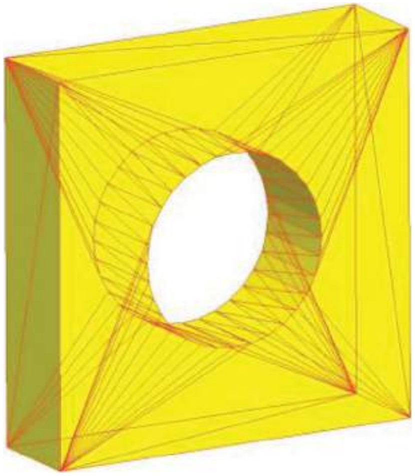


Figure 2. Triangular facets of a 3D CAD model.

softwares may use triangular facets to represent part models, and they provide the interface to get the corresponding vertex data of triangular facets. Fig. 2 gives triangular facets of a 3D CAD model. So, to analyze plane-plane contacts or cylinder- cylinder contacts, planes or surfaces usually are discretized into triangular facet meshes, and the vertices of triangular facets are chosen as initial sampling points. And line contacts sample on the tangents while point contacts are directly to capture contact points.

3.3. Contact status detection

Contact status detection is to check whether there is a common point between planes, surfaces, plane and surface on the premise that the prerequisite of contact relation exists. Here, we discuss three cases, detecting plane or surface contact, line contact and point contact.

3.3.1. Detecting plane or surface contact status

Generally, detecting contact status between planes or surfaces is to judge whether these planes or surfaces are fitting. And they can be classified into five categories (see Fig. 3): partial overlapping, inner overlapping, complete overlapping, cross overlapping and corner overlapping.

To detect the fitting between two planes, a plane is needed to sample. Subsequently, the sampled points are checked if they are common points of two planes. Because triangular facets can be used for representing part models in commercial CAD modeling softwares, CAD models are usually divided into triangular facets while their vertices are selected as the initial sampling

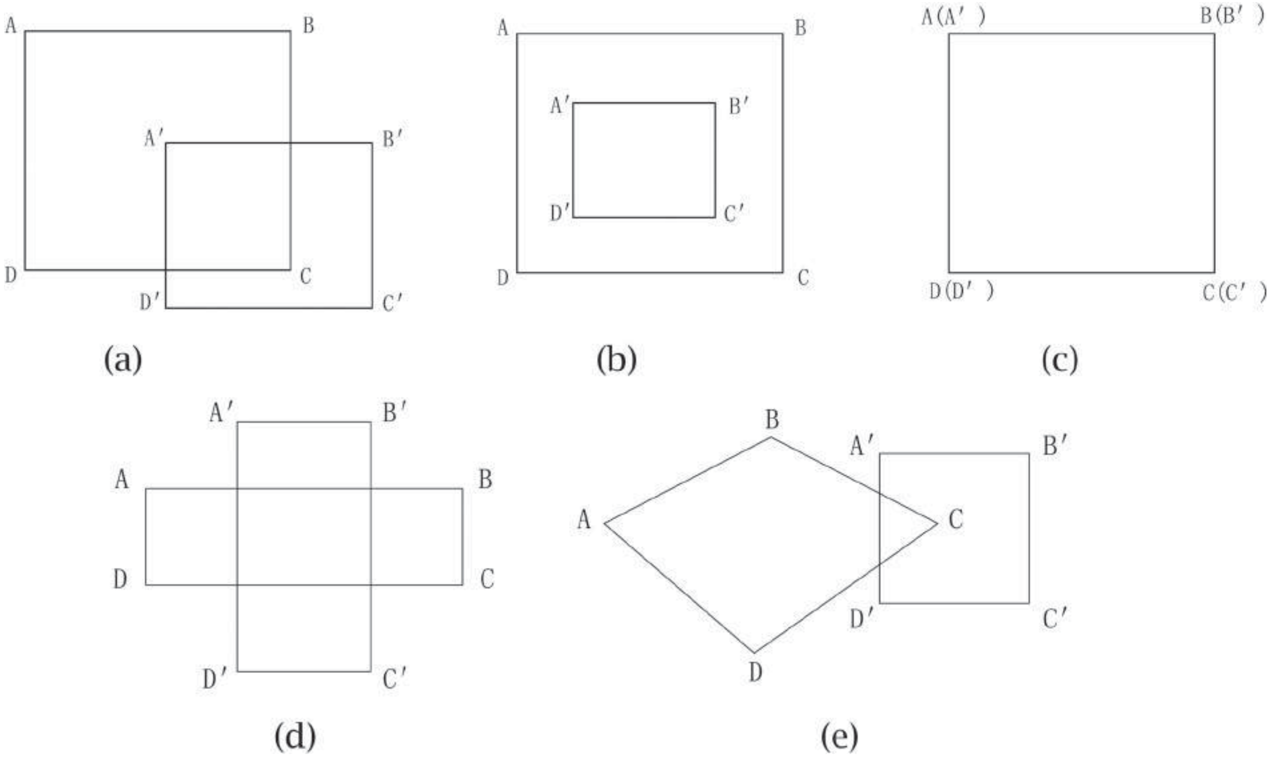


Figure 3. Contact status between planes or surfaces. (a) Partial overlapping, (b) Inner overlapping, (c) Complete overlapping, (d) Cross overlapping, and (e) Corner overlapping.

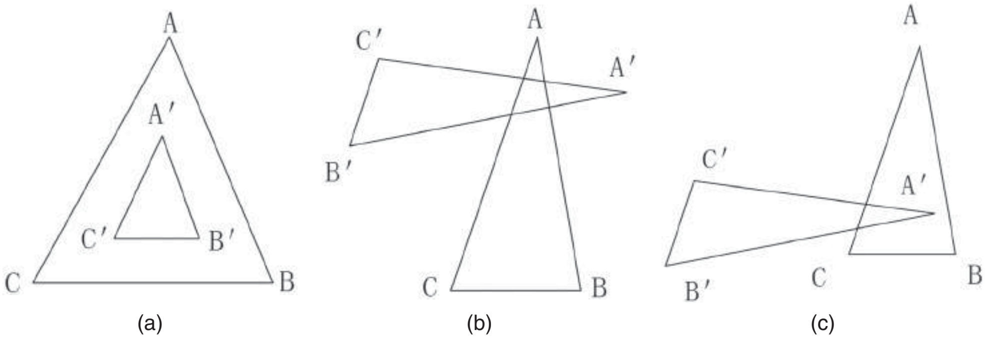


Figure 4. Contact relations of triangle elements. (a) Inner overlapping, (b) Cross overlapping, (c) corner overlapping.

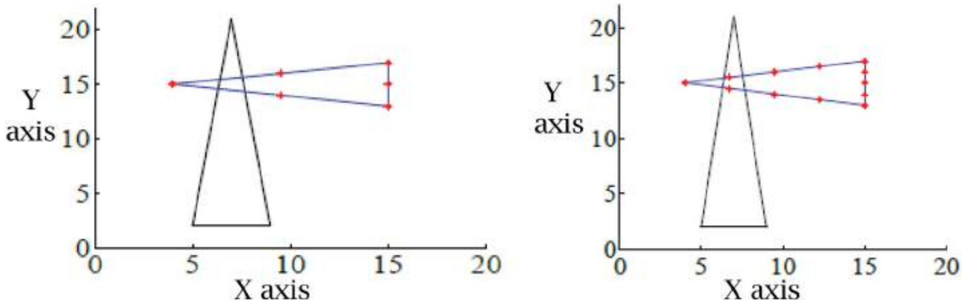


Figure 5. The dichotomy for segment sampling.

points. Different from finite element mesh generation, the vertices of triangular facets are on the boundaries of a model. So, initial sampling points apply only to the case of Fig. 3(a) while other cases are easy to be misjudged.

The cases of Figs. 3(a), (b) and (c) are universal, and Fig. 3(a) is easy to check using initial sampling points. But for Fig. 3(b), if an optional plane is initially sampled, the case of Fig. 4(a) might appear. At this point, sampling the smaller plane is appropriate, which has low complexity and accurate result. In Fig. 3(c), the midpoint of triangular facets' edge is chosen as initial sampling points. Then, compare the area of two planes and the smaller plane is sampled.

The cases of Figs. 3(d) and (e) are rare. Even if sample on the midpoint of triangular facets' edge, they might be misjudged (see the cases of Fig. 4(b) and Fig. 4(c)). Then, more sampling points are needed. The dichotomy is one of the best sampling ways for the boundaries of triangle facets.

For the case of Fig. 5, initial sampling might make mistakes. Then, the boundaries of the smaller triangular facet are divided using the dichotomy, until the boundaries of facets are partitioned into 2^n (n = 2, 3, ... , m) segments. The value of n is large, it has high computational complexity and precise. In general, the maximum value of n is 6.

The fitting between cylinders usually manifests as the mating between the axle hole and the shaft. Initial sampling for contact relation analysis in Figs. 3(a), (b) and (c) is feasible. Different from plane contacts, the midpoints of triangular facets' boundaries are not on a surface. Then, they are projected to the surfaces and the projected points are selected as sampling points (see Fig. 6).

Therefore, the procedures of detecting contact status between planes or surfaces can be reduced to the following two steps.

Step 1. Calculate the areas of judging planes or surfaces, and the smallest plane or surface is divided into triangular facets. Then, selecting the vertices of facets as the initial sampling points to judge whether there is a common point. If there is a common point, contact relation exists; else, go to the next step.

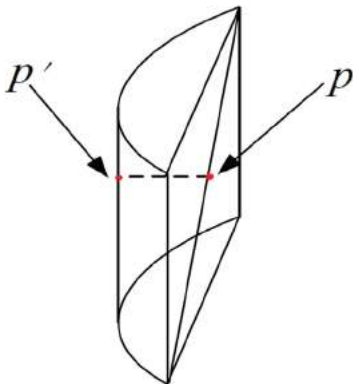


Figure 6. The projection of the boundary midpoint for a surface mesh element.

Step 2. Set the maximum value of n. If initial sampling point is unable to determine the contact status, the dichotomy is adopted to divide the existing triangular facets. Divide-points are selected as sampling points for planes. For surfaces, divide-points are projected to surfaces and the projected points are chosen as sampling points to judge whether they have common points. If they have, the judging surfaces are fitting; else, continue to divide until the boundaries of facets are divided into 2^n segments.

3.3.2. Detecting line contact status

Line contact usually manifests as the tangency between plane and cylinder. Fig. 7 gives three types of line contacts. The procedures for detecting line contact can be reduced to the following two steps.

Step 1. Calculate the coincidence generatrix of the tangency between plane and cylinder using geometric relationships. Then, judge whether the endpoints of the generatrix have a common point.

Step 2. If there is not a common point, set the maximum value of n. And the dichotomy is adopted to sample on the generatrix. If sampling points have a common point, the plane and cylinder are tangent; else, continue to divide the generatrix until it is divided into 2^n segments.

3.3.3. Detecting point contact status

Point contact usually appears as the tangency between spherical and plane (see Fig. 2(d)), the detection is relatively simple. The procedures for detecting point contact can be reduced to the following two steps.

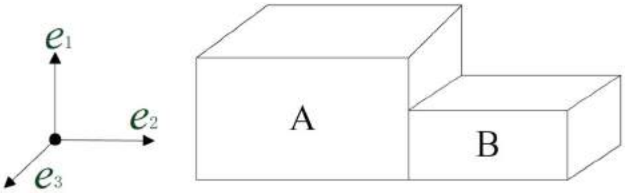


Figure 9. Spherical geometry method for the contact between two planes.

Step 1. Calculate the tangent point based on geometrical relationships between sphere and geometric plane.

Step 2. Judge whether the tangent point is on the plane. If the tangent point is on the plane, they are point touch; else, they don't belong to the point contact.

3.4. Contact relation expression

Contact relation pairs may be converted into constraint relations, which will restrict the movements of the parts in a certain direction. When all contact status is found, each part will have a set of corresponding constraint relations. Accordingly, these constraints ultimately determine local feasible moving direction of the parts. Therefore, we assume that an assembly model is composed of some solid parts and their corresponding constraint relations, which can be represented as a contact relation graph G = (P, C). Here, P describes a solid part and C denotes the relations between the parts. Figs. 8(a), (b) and (c) respectively give a hinge model with five parts, its corresponding contact relation graph and contact relation

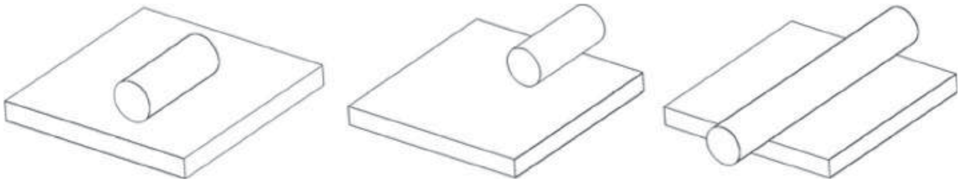
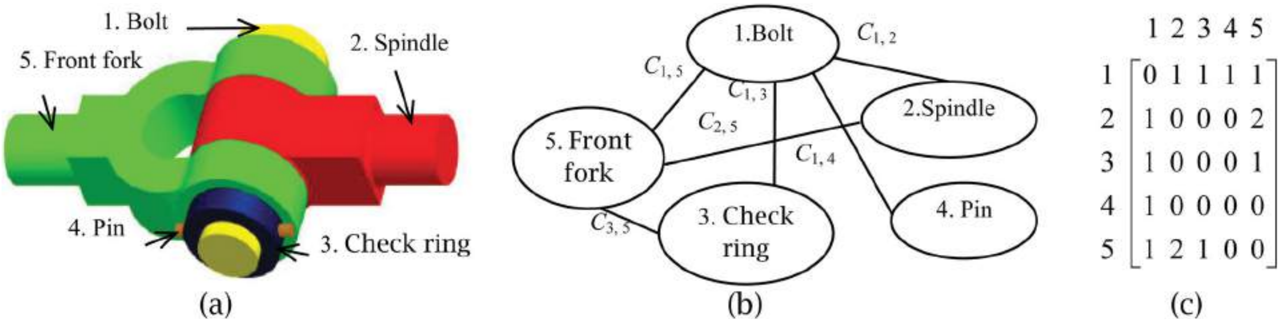


Figure 7. The tangency between plane and cylinder.



$$\begin{array}{c|ccccc} & 1 & 2 & 3 & 4 & 5 \\ \hline 1 & 0 & 1 & 1 & 1 & 1 \\ 2 & 1 & 0 & 0 & 0 & 2 \\ 3 & 1 & 0 & 0 & 0 & 1 \\ 4 & 1 & 0 & 0 & 0 & 0 \\ 5 & 1 & 2 & 1 & 0 & 0 \end{array}$$

Figure 8. A hinge model, its contact relation graph and contact relation matrix. (a) Hinge model, (b) Contact relation graph, and (c) Contact relation matrix.

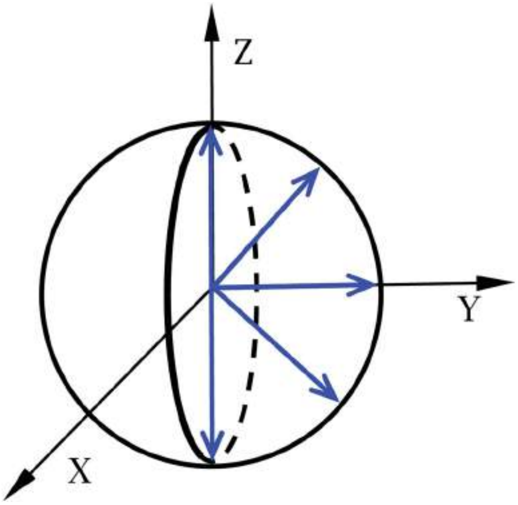


Figure 10. The disassembly directions mapped to Gauss sphere for part B in Fig. 9.

matrix. In Fig. 8(c), the elements of the matrix represent the number of contact surfaces. "0" denotes that there is no touch face or surface between two parts while "2" represents that two parts have two touch faces or surfaces.

4. The algorithm of generating ASP

4.1. Feasible assembly direction analysis for parts

To reduce the computational complexity, the existing methods usually define the disassembly direction of an assembly model as the parallel direction of an axle. But they might ignore and miss the actual disassembly direction. Then, spherical geometry method is adopted to analyze feasible assembly direction in this paper. The reason is that spherical geometry approach can represent all feasible movement directions of a part. The parts for the disassembling only have translation motion. The set of local feasible moving directions for no-constrained parts is regarded as a unit sphere. And a pair of contact relations will produce constraint relations between two parts. Then, the normal directions of local feasible movements for constrained parts are described as Gauss spheres. When all contact relations corresponding to Gauss spheres are calculated, the intersection of all Gauss spheres is the set of local feasible directions. Gauss sphere is defined as fellows.

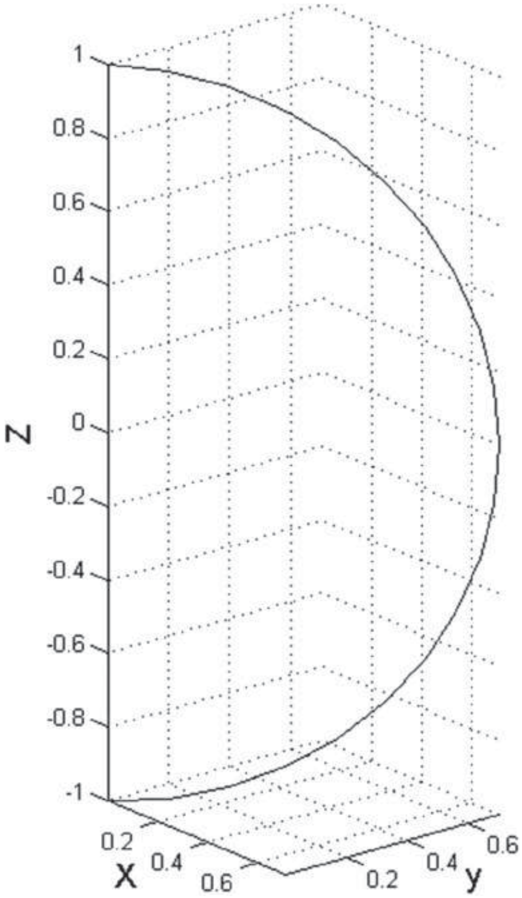


Figure 12. The set of feasible moving directions for a split sliding bearing

$$S = \{c = (x, y, z)|(x^2 + y^2 + z^2)^{1/2} = 1, (x, y, z) \in M\} \tag{4.1}$$

Here, M is the constraint condition of forming contact relations.

In Fig. 9, vectors $\boldsymbol{e_1}$, $\boldsymbol{e_2}$ and $\boldsymbol{e_3}$ form an assembly coordinate system, and parts A and B have a fitting plane.

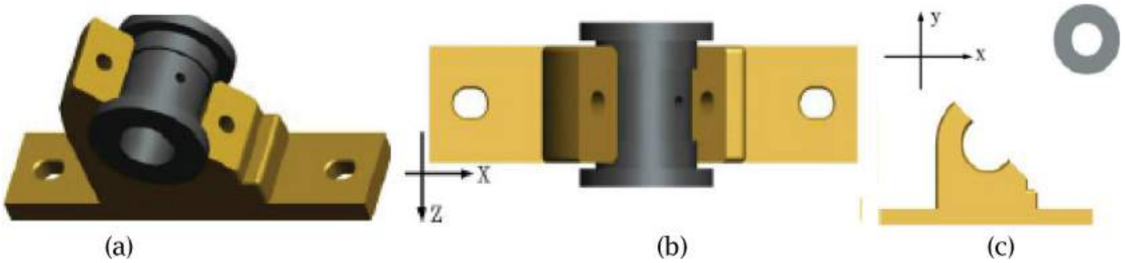


Figure 11. The structure of a split sliding bearing. (a) Axonometric drawing, (b) Front view, (c) The result of disassembling.

Table 1. The detecting path parameters for bearing in Fig. 9

ρ_1	π_1
1	0
-1	0
0	1
$\frac{1}{\sqrt{2}}$	$\frac{1}{\sqrt{2}}$
$-\frac{1}{\sqrt{2}}$	$\frac{1}{\sqrt{2}}$

Then, the set of local feasible direction for part A corresponding to Gauss spheres can be represented as:

$$S_b = \{c = (x, y, z)|(x^2 + y^2 + z^2)^{1/2} = 1, c.\boldsymbol{e_2} \leq 0\} \tag{4.2}$$

Namely, constraint condition $M = \{c.e_2 \leq 0\}$. The disassembly directions mapped to Gauss sphere for part B in Fig. 9 are given in Fig. 10.

Suppose part P has n fitting constraints, there are accordingly n constraints $M_1, M_2, \ldots, M_i, \ldots, M_n$. The constraint of Gauss spheres for part P can be described as $M_P = M_1 \cap M_2 \cap M_3 \ldots \cap M_i \ldots \cap M_n$. Therefore, a set of disassembly directions for part P corresponding to Gauss spheres can be represented as:

$$S_P = \{c = (x, y, z)|(x^2 + y^2 + z^2)^{1/2} = 1, (x, y, z) \in M_p\}$$

From the definition of the constraint condition M in Eqn. (4.2), we can conclude that the constraint M_i of each contact status forms an inequality. Therefore, constraint condition M_p has n inequalities. Then, solving Gauss spheres is to resolve the constraints M_p with n inequalities. If the solution set is null, it denotes that there is no feasible moving direction and this part is disassembled. When the solution set is not empty, a vector basis is used for representing a set of local feasible moving directions for a part.

4.2. Detecting local feasible moving directions

For the parts with local feasible moving directions, a set of moving directions of Gauss spheres form a polyhedral convex cone. Then, a polyhedral convex cone can be defined as follows:

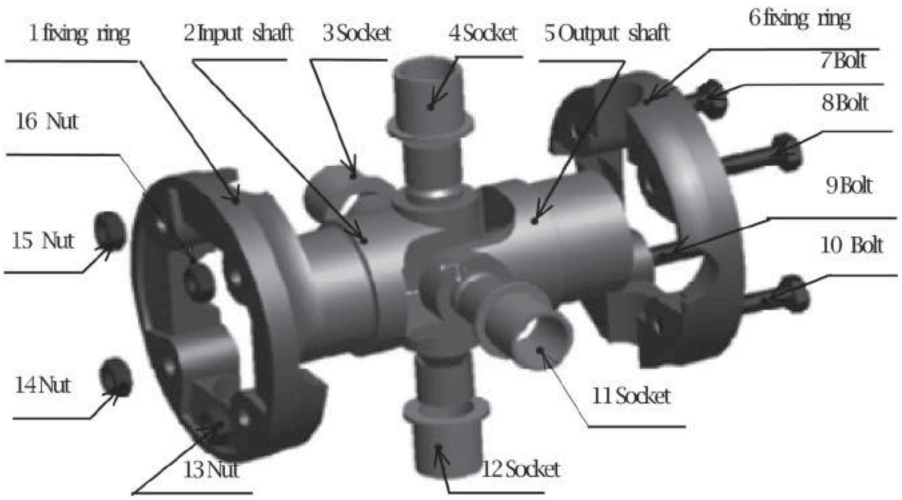


Figure 13. A cross coupling model. (a)The complete disassembly of a cross coupling, (b) The layers of ASP.

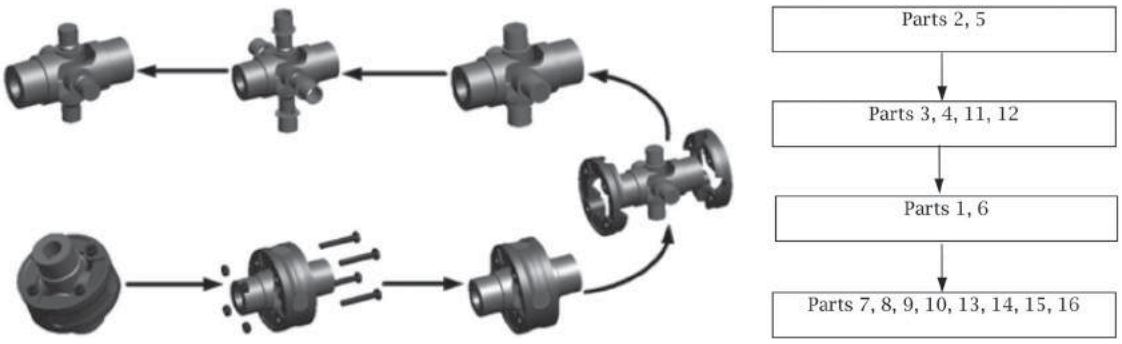


Figure 14. The diaassembly result of a cross coupling. (a)The complete disassembly of a cross coupling, (b) The layers of ASP.

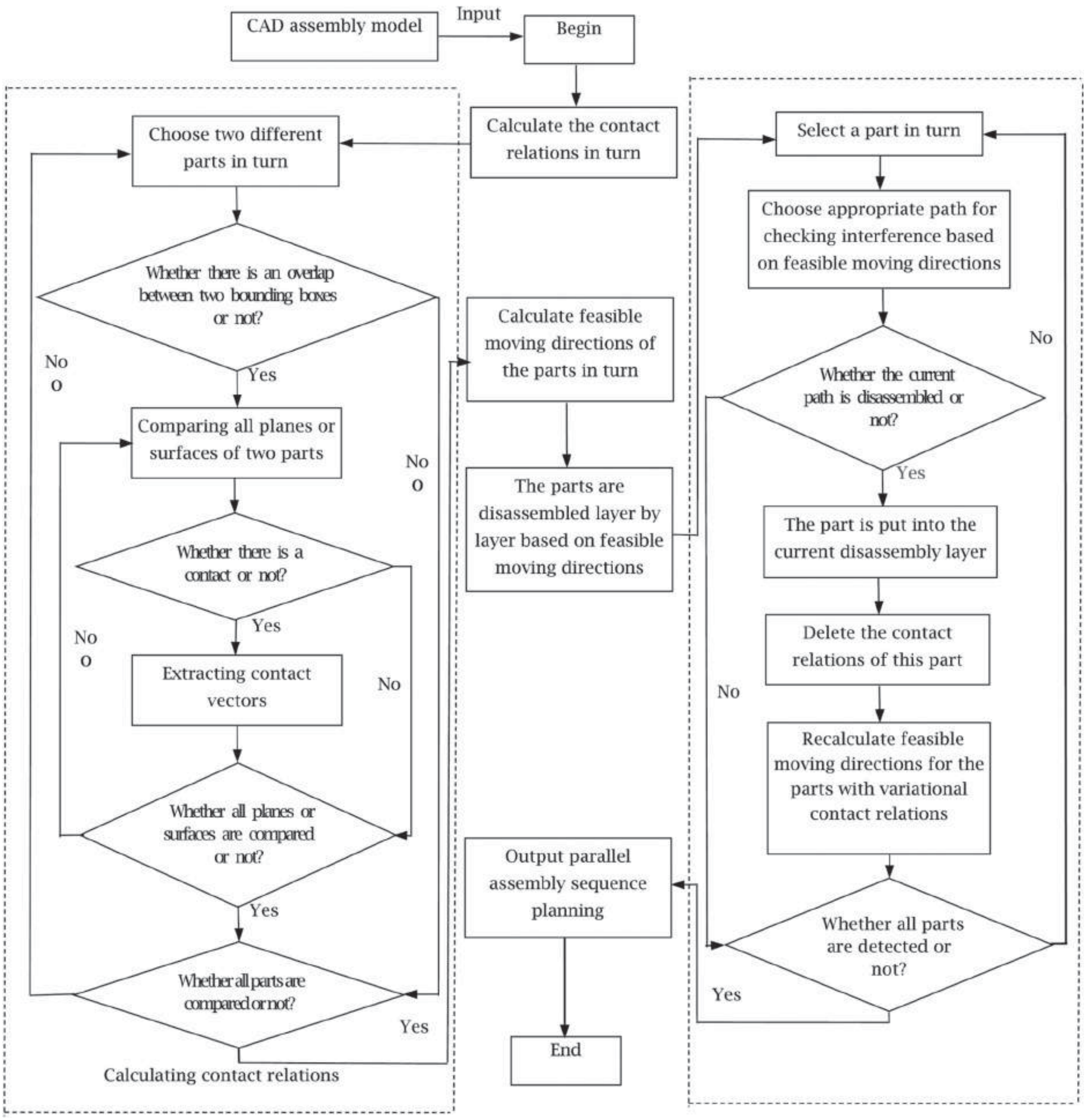

CAD assembly model
Input
Begin
Calculate the contact relations in turn
Choose two different parts in turn
Whether there is an overlap between two bounding boxes or not?
Yes
No o
Comparing all planes or surfaces of two parts
Whether there is a contact or not?
Yes
No
Extracting contact vectors
No o
Whether all planes or surfaces are compared or not?
Yes
No o
Whether all parts are compared or not?
Yes
Calculating contact relations
Calculate feasible moving directions of the parts in turn
The parts are disassembled layer by layer based on feasible moving directions
Output parallel assembly sequence planning
End
Select a part in turn
Choose appropriate path for checking interference based on feasible moving directions
Whether the current path is disassembled or not?
Yes
No
The part is put into the current disassembly layer
Delete the contact relations of this part
Recalculate feasible moving directions for the parts with variational contact relations
Whether all parts are detected or not?
Yes
No

Figure 15. The flow diagram of generating ASP.

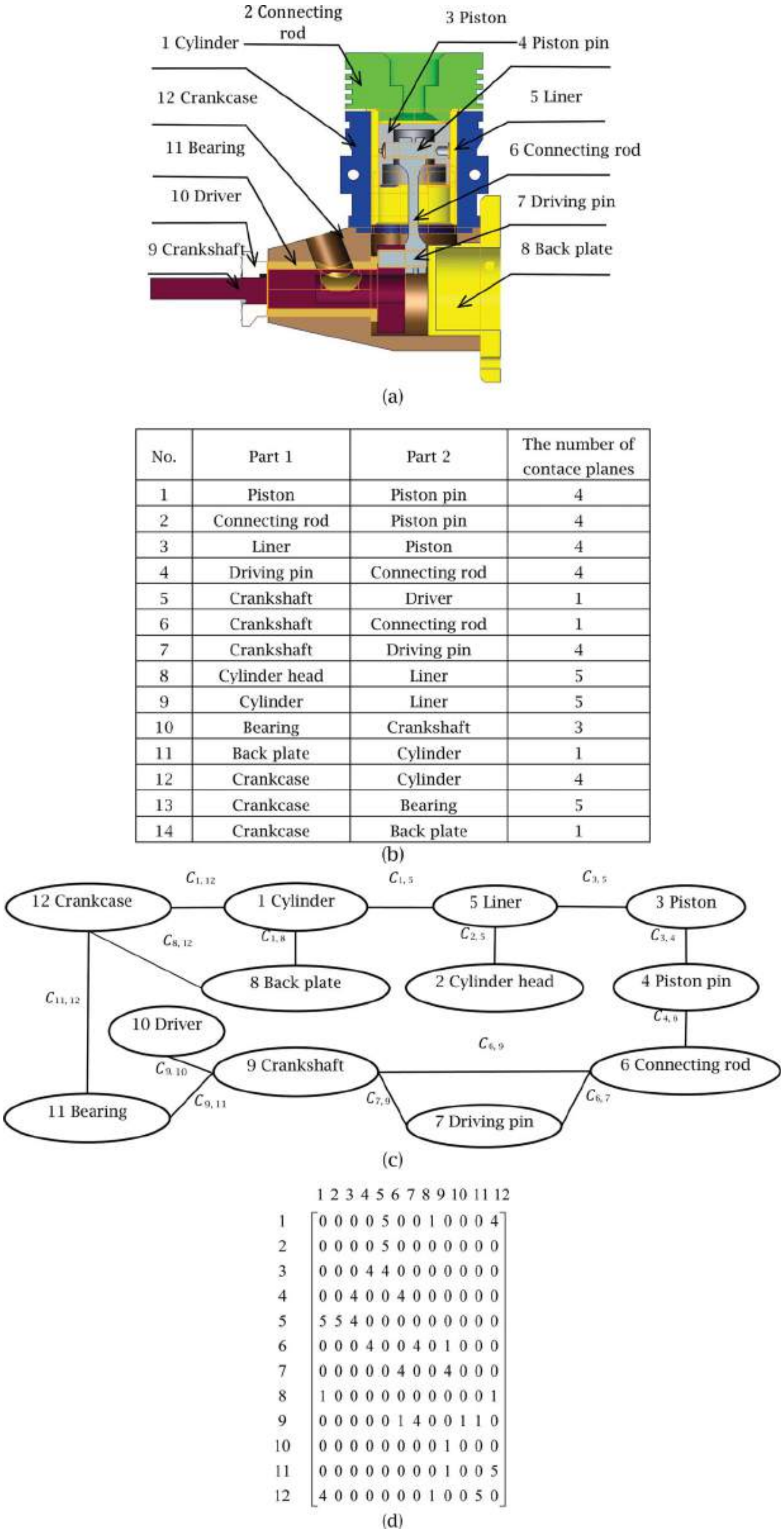
Definition 1. The general expression of a polyhedral convex cone: Given a cone C$_\pi$, its basis vector{C$_1$, C$_2$, ... , C$_\text{n}$} can be divided into two cases:

(1) The opposite basis vectors belong to cone C$_\pi$, $A \equiv \{c_i| - c_i \in C_\pi\}$;
(2) The opposite basis vectors don't belong to cone C$_\pi$,$B \equiv \{c_i| - \boldsymbol{c}_i \notin C_\pi\}$. Therefore, the general expression of C$_\pi$ can be represented as follows:

$$C_\pi \equiv (A| - A|B)_\pi \equiv A_\rho + B_\pi.$$

The Γ algorithm [2] is adopted to solve linear inequality equations and a set of basis vectors for cone C$_\pi$ is found. The basis vectors with equality are categorized as $c_j \in A_\rho$ while the basis vectors with inequality are categorized as $c_k \in A_\pi$. And C$_\pi$ can be expressed as:

$$\begin{aligned} C_\pi &= \rho_1 c_1 + \rho_2 \boldsymbol{c}_2 + \ldots + \rho_m \boldsymbol{c}_m + \pi_{m+1} \boldsymbol{c}_{m+1} \\ &\quad + \pi_{m+2} \boldsymbol{c}_{m+2} + \ldots + \pi_n \boldsymbol{c}_n; \\ &\rho_j \in R, \pi_k \in R^+; j = 1, 2, \ldots, m; \\ &k = m+1, m+2, \ldots, n. \end{aligned}$$



(a)

No.	Part 1	Part 2	The number of contace planes
1	Piston	Piston pin	4
2	Connecting rod	Piston pin	4
3	Liner	Piston	4
4	Driving pin	Connecting rod	4
5	Crankshaft	Driver	1
6	Crankshaft	Connecting rod	1
7	Crankshaft	Driving pin	4
8	Cylinder head	Liner	5
9	Cylinder	Liner	5
10	Bearing	Crankshaft	3
11	Back plate	Cylinder	1
12	Crankcase	Cylinder	4
13	Crankcase	Bearing	5
14	Crankcase	Back plate	1

(b)

(c)

$$\begin{array}{c} \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \\ 6 \\ 7 \\ 8 \\ 9 \\ 10 \\ 11 \\ 12 \end{array}
\begin{array}{c} \begin{array}{cccccccccccc} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 & 11 & 12 \end{array} \\
\begin{bmatrix}
0 & 0 & 0 & 0 & 5 & 0 & 0 & 1 & 0 & 0 & 0 & 4 \\
0 & 0 & 0 & 0 & 5 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 4 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 4 & 0 & 0 & 4 & 0 & 0 & 0 & 0 & 0 & 0 \\
5 & 5 & 4 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 4 & 0 & 0 & 4 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 & 4 & 0 & 0 & 0 \\
1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0 & 0 & 0 & 0 & 1 & 4 & 0 & 0 & 1 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 5 \\
4 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 5 & 0
\end{bmatrix} \end{array}$$

(d)

Figure 16. Contact relation analysis for a motor. (a) Highlights for contact planes or surfaces, (b) The table of contact relations, (c) Contact relation graph, (d) Contact relation matrix.

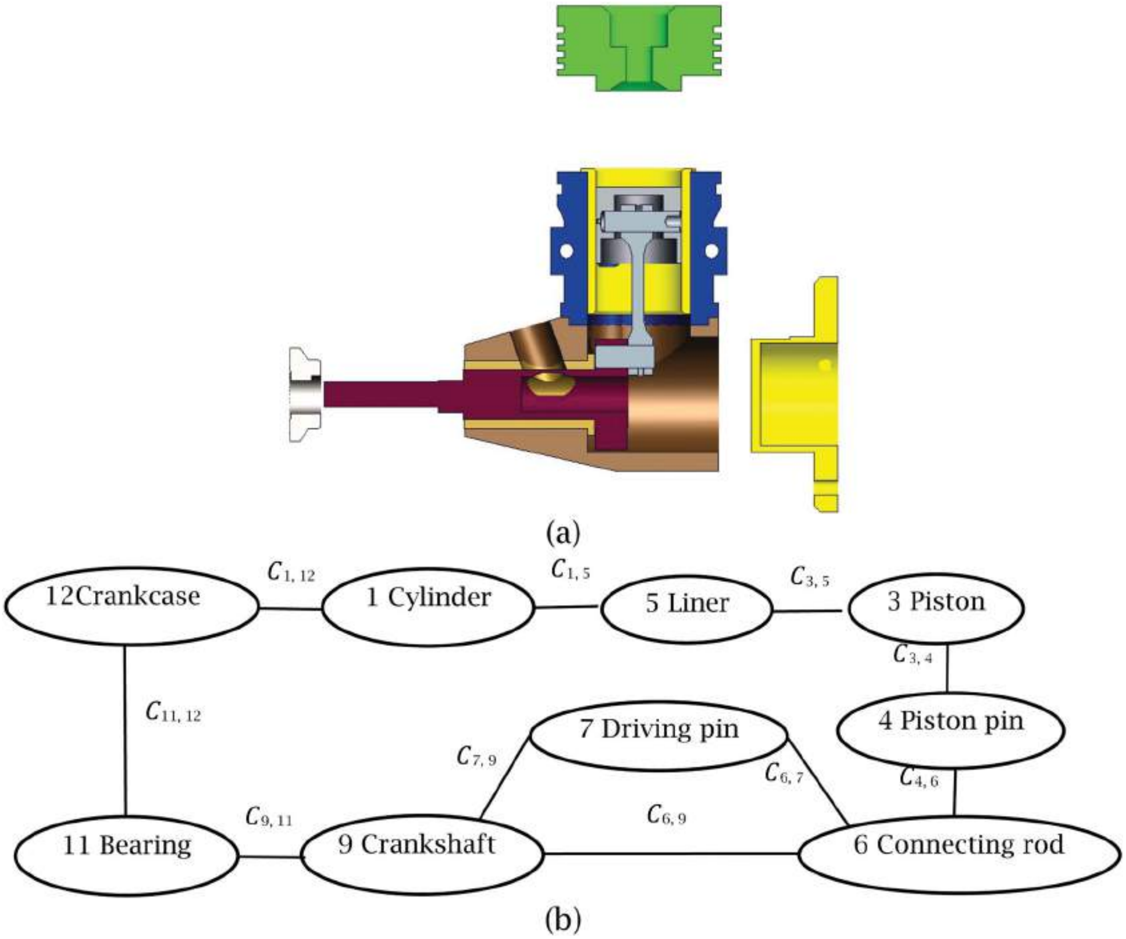


Figure 17. The first disassembling. (a)The first cycle of disassembling, (b) Contact relation graph of remainder parts.

So, detecting local feasible moving directions is to determine the coefficient of the cone C_π. Fig. 9 gives the structure of a split sliding bearing. According to the definition of vectors e_1, e_2 and e_3 in Fig. 9, contact relation analysis and assembly coordinate system in Fig. 11(b), a split sliding bearing in Fig. 11(a) has the following constraints.

$$M_1 = \{\boldsymbol{c} | \boldsymbol{c} \cdot \boldsymbol{e}_1 \leq 0\}, \boldsymbol{e}_1 = \begin{bmatrix} -\frac{1}{\sqrt{2}} & \frac{1}{\sqrt{2}} & 0 \end{bmatrix}^T;$$

$$M_2 = \{\boldsymbol{c} | \boldsymbol{c} \cdot \boldsymbol{e}_2 \leq 0\}, \boldsymbol{e}_2 = \begin{bmatrix} \frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \end{bmatrix}^T;$$

$$M_3 = \{\boldsymbol{c} | \boldsymbol{c} \cdot \boldsymbol{e}_3 \leq 0\}, \boldsymbol{e}_3 = \begin{bmatrix} -\frac{1}{\sqrt{2}} & -\frac{1}{\sqrt{2}} & 0 \end{bmatrix}^T;$$

$$M = M_1 \cap M_2 \cap M_3.$$

Therefore, the constraints M form the following inequality matrix equations.

$$C\boldsymbol{x} = \begin{bmatrix} \boldsymbol{e}_1 & \boldsymbol{e}_2 & \boldsymbol{e}_3 \end{bmatrix}^T \boldsymbol{x} \leq 0.$$

Then, the cone C_π has the following basis vectors.

$$C_\pi = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix}_\rho + \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \\ 0 \end{bmatrix}_\pi = \rho_1 \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} + \pi_1 \begin{bmatrix} 1/\sqrt{2} \\ 1/\sqrt{2} \\ 0 \end{bmatrix}; \rho_1 \in R, \pi_1 \in R^+.$$

Thus, the split sliding bearing has the following set of feasible moving directions (see Fig. 12). Because basis vectors are the boundaries of a cone C_π and satisfy the constraints, they may sever as an initial moving direction to detect. Then, the next detecting path is uniformly chosen on the cone. To reduce the computational complexity, the detecting path should not be too much. For the example above, the values of interval $[\rho_1, \pi_1]$ in above example are listed in Tab. 1. And the result of disassembly is given in Fig. 11(c).

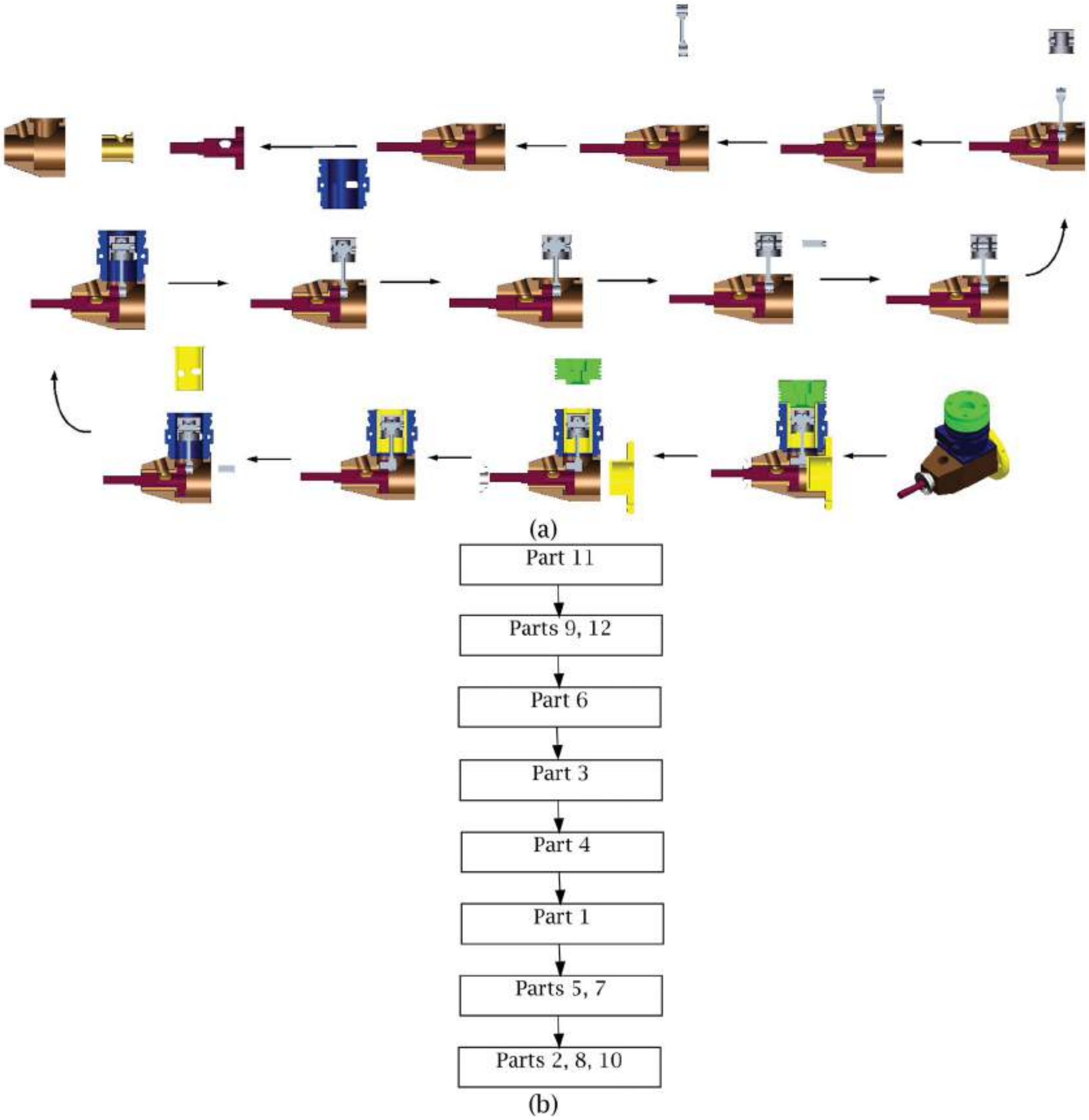


Figure 18. The result of disassembling of a motor. (a) The complete disassembly of a motor, (b) The precedence constraint graph of AS.

4.3. Generating ASP

The assemblies are usually considered to be a serial process in the existing literature. Thus, the parts of an assembly model are assembled one by one for generating ASP. So, the actual assembly and generated assembly planning might be inconsistent. And sorting assembly planning caused feasible AS is increasing with the number of the parts.

For a given assembly model, the parts for parallel assembling are determined in a state. At this point, these parallel parts for assembling are no sequence in geometry. And AS is determined by its assembly process. Therefore, an assembly model can produce the only sequence planning based on the analysis of assembly layers.

In Fig. 13, an assembly model called cross coupling is listed to explain generating ASP in parallel. There are 16 parts in a cross coupling. Firstly, calculate feasible moving directions of each part. Then, select the appropriate moving directions from the parts with non-empty set of feasible moving directions, and detect whether these directions have interferences. If the interference doesn't exist, these parts can be disassembled in a cycle. The complete disassembly of a cross coupling is given in Fig. 14(a). In a cycle, disassembled parts and their contact relations are removed from the contact relation graph, until all parts are disassembled and their contact relations are deleted. The layers of ASP for a cross coupling is listed in Fig. 14(b). The first layer has input shaft 2 and output shaft 5, sockets 3, 4, 11 and 12 are disassembled in the second layer while fixing rings 1 and 6 are removed from the third layer. Finally, the bolts 7, 8, 9, 10 and the nuts 13, 14, 15, 16 are disassembled in the fourth layer. The assembly and the disassembly are opposite.

Based above analysis, the flow diagram of generating ASP is given in Fig. 15.

5. An example for the proposed approach

In this paper, the development tool Pro/Toolkit of Pro/Engineer and Visual Studio are used to generate ASP automatically. A motor with 12 parts is introduced to illustrate the proposed approach. Firstly, extract contact relations of parts from assembly models' geometry information. In Fig. 16(a), the highlights represent contact planes or surfaces of a motor. Then, generate contact relation table (see Fig. 16(b)), contact relation graph (see Fig. 16(c)) and contact relation matrix (see Fig. 16(d)) based on contact relations of parts. The computing time of contact relations is 0.25 s.

The contact relations between parts form the constraints. Firstly, calculate a set of local feasible moving directions for each part by solving inequality equations. Secondly, the parts with non-empty direction set are selected to check interference on a given path for finding feasible disassembly parts. Fig. 17(a) gives the result of the first disassembling, cylinder head 2, back plate 8 and driver 10 lie in the first layer. And Fig. 17(b) lists contact relation graph of the reminder parts except for parts 2, 8 and 10. Meanwhile, the related data of parts 2, 8 and 10 is removed. Repeat the process above until there is a part or no part in a contact relation graph. The complete disassembly of a motor is given in Fig. 18 (a) and the layer graph of AS for a motor is listed in Fig. 18 (b). It can be seen from the layer graph of AS that assembly model has been disassembled into 7 layers. In an assembly layer, there is no sequence between parts, and the parts can be assembled in any order.

6. Conclusion and future works

The proposed approach extracts contact relations of parts from assembly models, and the constraint between parts is determined by these contact relations. Then, feasible moving directions of each part are calculated based on the constraints between parts. To find feasible disassembled parts, translating paths are built and interferences are checked in feasible directions. Repeating the process above, assembly models are transformed into layer graphs of AS. The proposed method has an advantage in improving the automation of ASP. And the problem that automatically selects the directions of three dimensional non-orthogonal assembling is resolved. In the future, we would be a focus for the work into verifying as well as optimizing the feasibility of ASP. For the feasibility of assembly planning, some factors like assembly tools and connection types of parts should be taken into account. Meanwhile, some intelligent algorithms should be introduced to optimize the assembly efficiency.

ORCID

Songqiao Tao http://orcid.org/0000-0002-5799-4670

References

[1] Bourjault, A.: Contribution á une approche méthodologique de l'assemblage automatisé: Elaboration automatique des séquences opératoires, Ph.D. Thesis, Faculté des Sciences et des Techniques de l'Université de Franche-Comté, France.

[2] Castillo, E.; Jubete, F.; Pruneda, E.; et al.: Obtaining simultaneous solutions of linear subsystems of equations and inequalities, Linear Algebra and its Applications, 346(1–3), 2002, 131–154. https://doi.org/10.1016/S0024-3795(0100500-6)

[3] De Fazio, T.; Whitney, D.: Simplified generation of all mechanical assembly sequences, IEEE Journal on Robotics and Automation, 3(6), 1987, 640–58. https://doi.org/10.1109/JRA.1987.1087132

[4] Dong, T.; Tong, R.; Zhang, L.; et al.: A knowledge-based approach to assembly sequence planning, The International Journal of Advanced Manufacturing Technology, 32(11–12), 2007, 1232–1244. https://doi.org/10.1007/s00170-006-0438-1

[5] Ghandi, S.; Masehian, E.: Assembly sequence planning of rigid and flexible parts, Journal of Manufacturing Systems, 36(7), 2015, 128–146. https://doi.org/10.1016/j.jmsy.2015.05.002

[6] Ghandi, S.; Masehian, E.: Review and taxonomies of assembly and disassembly path planning problems and approaches, Computer-Aided Design, 67, 2015, 58–86. https://doi.org/10.1016/j.cad.2015.05.001

[7] Gottipolu, R.; Ghosh, K.: Representation and selection of assembly sequences in computer-aided assembly process planning, International journal of production research, 35(12), 1997, 3447–3466. https://doi.org/10.1080/002075497194183

[8] Gottipolu, R.; Ghosh, K.: A simplified and efficient representation for evaluation and selection of assembly sequences, Computers in Industry, 50(3), 2003, 251–264. https://doi.org/10.1016/S0166-3615(03)00015-0

[9] Gruhier, E.; Demoly, F.; Dutartre, O.; et al.: A formal ontology-based spatiotemporal mereotopology for integrated product design and assembly sequence planning, Advanced Engineering Informatics, 29(3), 2015, 495–512. https://doi.org/10.1016/j.aei.2015.04.004

[10] Gu, T.; Xu, Z.; Yang, Z.: Symbolic OBDD representations for mechanical assembly sequences, Computer-Aided Design, 40(4), 2008, 411–421. https://doi.org/10.1016/j.cad.2007.12.001

[11] Henrioud, J.; Bourjault, A.: LEGA: a computer-aided generator of assembly plans, Computer-Aided Mechanical Assembly Planning, Springer US, 1991, 191–215. https://doi.org/10.1007/978-1-4615-4038-0_8

[12] Hsu, Y.; Tai, P.; Wang, M.: A knowledge-based engineering system for assembly sequence planning, The International Journal of Advanced Manufacturing Technology, 55(5–8), 2011, 763–782. https://doi.org/10.1007/s00170-010-3093-5

[13] Huang, Y.; Lee, C.: A framework of knowledge-based assembly planning, 1991 IEEE International Conference on Robotics and Automation, IEEE, 1991. https://doi.org/10.1109/ROBOT.1991.131647

[14] Jiménez, P.: Survey on assembly sequencing: a combinatorial and geometrical perspective, Journal of Intelligent Manufacturing, 24(2), 2013, 235–250. https://doi.org/10.1007/s10845-011-0578-5

[15] Jiménez, P.: Survey on assembly sequencing: a combinatorial and geometrical perspective, Journal of Intelligent Manufacturing, 24(2), 2013, 235–250. https://doi.org/10.1007/s10845-011-0578-5

[16] Kashkoush, M.; Elmaraghy, H.: Knowledge-based model for constructing master assembly sequence, Journal of Manufacturing Systems, 34(1), 2015, 43–52. https://doi.org/10.1016/j.jmsy.2014.10.004

[17] Kavraki, L.; Latombe, J.; Wilson R.: On the complexity of assembly partitioning, Information Processing Letters, 48(5), 1993, 229–235. https://doi.org/10.1016/0020-0190(9390085-N)

[18] Laperriére, L.; ElMaraghy, H.: GAPP: a generative assembly process planner, Journal of Manufacturing systems, 15(4), 1996, 282. https://doi.org/10.1016/0278-6125(96)84553-5

[19] Lin, M.; Tai, Y.; Chen, M.; et al.: A rule based assembly sequence generation method for product design, Concurrent Engineering, 15(3), 2007, 291–308. https://doi.org/10.1177/1063293X07083084

[20] Ou, L.; Xu, X.: Relationship matrix based automatic assembly sequence generation from a CAD model, Computer-Aided Design, 45(7), 2013, 1053–1067. https://doi.org/10.1016/j.cad.2013.04.002

[21] Pan, C.: Integrating CAD files and automatic assembly sequence planning (Doctoral dissertation), Iowa State University, 2005.

[22] Qu, S.; Jiang, Z.; Tao, N.: An integrated method for block assembly sequence planning in shipbuilding, The International Journal of Advanced Manufacturing Technology, 69(5), 2013, 1123–1135. https://doi.org/10.1007/s00170-013-5087-6

[23] Raghavan, V.; Molineros, J.; Sharma, R.: Interactive evaluation of assembly sequences using augmented reality, IEEE Transactions on Robotics and Automation, 15(3), 1999, 435–449. https://doi.org/10.1109/70.768177

[24] Rashid, M.; Hutabarat, W.; Tiwari, A.: A review on assembly sequence planning and assembly line balancing optimisation using soft computing approaches, The International Journal of Advanced Manufacturing Technology, 59(1–4), 2012, 335–349. https://doi.org/10.1007/s00170-011-3499-8

[25] Santochi, M.; Dini, G.: Computer aided planning of assembly operations: the selection of assembly sequences, Robotics and Computer-Integrated Manufacturing, 9(6), 1992, 439–46.

[26] Son, C.: Intelligent rule-based sequence planning algorithm with fuzzy optimization for robot manipulation tasks in partially dynamic environments, Information Sciences, 342, 2016, 209–221. https://doi.org/10.1016/j.ins.2015.08.020

[27] Wang, J.; Liu, J.; Zhong, Y.: A novel ant colony algorithm for assembly sequence planning, The International Journal of Advanced Manufacturing Technology, 25(11–12), 2005, 1137–1143. https://doi.org/10.1007/s00170-003-1952-z

[28] Wang, Z.; Ong, S.; Nee, A.: Augmented reality aided interactive manual assembly design, The International Journal of Advanced Manufacturing Technology, 69(5–8), 2013, 1311–1321. https://doi.org/10.1007/s00170-013-5091-x

[29] Wang, Y.; Liu, J.: Chaotic particle swarm optimization for assembly sequence planning, Robotics and Computer-Integrated Manufacturing, 26(2), 2010, 212–222. https://doi.org/10.1016/j.rcim.2009.05.003

[30] Wilson, R.: Minimizing user queries in interactive assembly planning, IEEE transactions on robotics and automation, 11(2), 1995, 308–312. https://doi.org/10.1109/70.370514

[31] Zhang, H.; Liu, H.; Li, L.: Research on a kind of assembly sequence planning based on immune algorithm and particle swarm optimization algorithm, The International Journal of Advanced Manufacturing Technology, 71(5), 2014, 795–808. https://doi.org/10.1007/s00170-013-5513-9

[32] Zhang, W.; Ma, M.; Li, H.; et al.: Generating interference matrices for automatic assembly sequence planning, The International Journal of Advanced Manufacturing Technology. https://doi.org/10.1007/s00170-016-9410-x

[33] Zhang, Y., Ni, J.; Lin, Z.; et al.: Automated sequencing and sub-assembly detection in automobile body assembly planning, Journal of materials processing technology, 129(1), 2002, 490–494. https://doi.org/10.1016/S0924-0136(02)00621-0