Disassembly Path Generation to Verify the Assemblability of Mechanical Products*

Eiji ARAI**, Naoki UCHIYAMA***
and Masanori IGOSHI****

A system for verifying assemblability is required in CAD/CAM systems for mechanical products, by which the shapes of parts can be modified to enable easy machining and assembly on the factory floor. This paper introduces a part assemblability verification system. Substituting the assemblability with disassemblability, a system that generates paths for disassembly is constructed based on a solid modeling system for mechanical products. The developed system takes contact constraint and contact transition into consideration. Algorithms to derive possible motions of parts which maintain contact states, and to calculate the positions and orientations (configurations) where contact states change are presented. An assemblability verifying method is proposed that uses tree structural data whose arcs and nodes denote possible motions, configurations respectively. The efficiency of the developed system for design verification and assembly planning is demonstrated through an example.

Key Words: Computer-Aided Design (CAD), Computer-Aided Manufacturing (CAM), Design, Assembly, Assembly Planning

1. Introduction

A system for verifying assemblability is required in CAD/CAM systems for mechanical products, by which the shapes of parts can be modified to enable easy machining and assembly on the factory floor. The assemblability of mechanical products must be verified from various points of view, such as availability of existing assembly facilities and geometrical interferences between part and assembly machine during assembly operation. One of the most essential viewpoints is the geometrical interferences among parts of the assembled product during assembly, because no assembly machine can assemble the product if it cannot be assembled due to its shape.

For verifying the assemblability of the product, verifying the assemblability of each part is necessary, since the product can be assembled if all parts can be assembled. An assemblability verification method for a part is also effective for the determination of assembly sequences for the product. We can obtain an assembly sequence among parts without geometrical interference by assembling an assemblable part preferentially.

Several studies concerning the assemblability verification method have been performed. For example, Kitajima et al. have developed an interactive system to verify the assemblability of a product[1]. Woo and Dutta have proposed a method to check the assemblability of a part for the assembly sequence generation[2]. However most of these methods restrict the assembly motion of a part to one translational motion only.

When we consider a product whose assembly operation is the reverse of its disassembly operation, finding a path for disassembly of a part without

* Received 11th March, 1994. Japanese original: Trans. Jpn. Soc. Mech. Eng., Vol. 59, No. 560, C (1993), pp. 1300-1307 (Received 13th May, 1992)

** Department of Welding and Production Engineering, Osaka University, 2-1 Yamadaoka, Suita, Osaka 565, Japan

*** Department of Mechanical Engineering, Toyohashi University of Technology, 1-1 Tempaku-cho, Toyohashi, Aichi 441, Japan

**** Department of Precision Engineering, Tokyo Metropolitan University, 1-1 Minami-osawa, Hachioji, Tokyo 192-03, Japan

colliding with the other parts means verifying its assemblability.

In this paper we propose a method to generate the disassembly path for each part of the product on the assumption that the assembled product is given, and assemblability can be verified by finding the disassembly path.

2. Method Overview

Most of the proposed methods for finding a collision-free path can be classified into two types: one is based on the configuration space approach[3], the other is based on the use of artificial potential[4]. However, these methods seem to be unsuitable for the disassemblability verification for the following reasons.

(1) From the viewpoint of verifying the disassemblability, a generated path does not have to be optimized with respect to any function. Our purpose is only to check whether a path exists without collision.

(2) There is more than one destination of a verified part. There are many configurations where disassembly of a part is terminated.

(3) Possible motion of each part in the product is limited, since it comes into contact with the other parts.

Studies concerning derivation of geometrically admissible small displacement from contact states have been done[5],[6], and are effective for the generation of disassembly path. However, non-small displacement is required to remove a part. For this, a collision detection method is necessary. Combining the small displacement derivation method and the collision detection method effectively allows us to develop the disassemblability verification method.

The developed method generates the disassembly path in the following manner. First, we calculate the possible translational and rotational motions of the verified part from its contact states in the product. The obtained motions keep the verified part in contact with the other parts. Next, the configurations where contact state transition occurs are calculated. Then the possible motions are calculated again from the updated contact states. By repeating these procedures, a problem of checking the disassemblability can be substituted with that of searching a path by traversing a tree whose nodes and arcs denote the configurations and the possible motions of a verified part, respectively. Finding a configuration where disassembly is terminated by traversing a tree means that a part is disassemblable. According to this method, a part is constrained from its contact states during disassembly; therefore, we can calculate the possible motions at any stage of the disassembly operation. This method cannot verify the disassemblability rigorously. However, its practical utility is high since mechanical parts are generally assembled without complex motion.

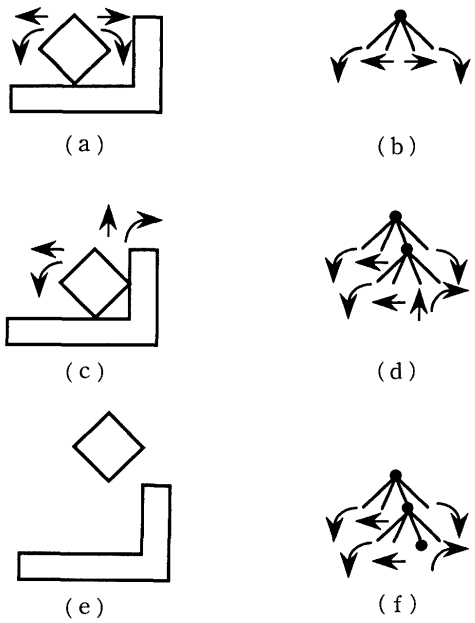

Fig. 1 Representation of disassembly path

Figure 1 shows the possible motions of a part and its tree description. The possible motions which keep a verified part in contact with other parts are calculated as shown in Fig. 1(a). This state can be represented by a tree as shown in Fig. 1(b). Then the part changes its contact states due to one of the possible motions, and again possible motions are calculated as shown in Figs. 1(c) and 1(d). Disassembly is terminated by translating a part using one of the motions calculated from the state shown in Fig. 1(c). Figures 1(e) and 1(f) show the finished disassembled state. In the following section, the details of this method are described.

3. Disassembly Path Generation

3.1 Geometric model and contact state

We assume that an assembled product is given by the CAD system and parts have rigid bodies. Let P and p be a verified part and one of its shape elements, respectively. Q and q denote all other parts excepting P and one of their shape elements.

When a product is represented by a polyhedron, the shape element is either a vertex, an edge or a planar surface (face). Axial parts with rotational functionality are typical in mechanical products. For this, we deal with cylindrical surfaces whose two ends are both circles. Their radius, normalized vector of the center axis, and position vector of a point on the axis, are given by the CAD system.

Contact states can be represented by the combination of the elements mentioned above. In our research, the contact states including a cylindrical surface are dealt with specifically. We regard only the case satisfying the following conditions as a cylindrical contact as shown in Fig. 2:

(1) Contacting elements are both cylindrical surfaces.

(2) Their axes are colinear.

(3) Their radii are the same.

Other contact states including a cylindrical surface are represented as those between polyhedra by polyhedral approximation of the cylindrical surface.

3.2 Generation of possible motions from contact states

The motion of a verified part is constrained by contact with the other parts. In our research, spatial motion is separated into translational motion and rotational motion. This separation allows analytical calculation of the configuration where contact states change as described in Section 4.

In Fig. 3, p and q, a vertex of a verified part P and a face of Q respectively, are in contact. Possible translational directions of P are represented by $\boldsymbol{x}\in \boldsymbol{R}^3$ which satisfies the following condition:

$$\boldsymbol{f}^T\boldsymbol{x}\geq 0$$

where $\boldsymbol{f}$ denotes the outward normal vector of face q. Moreover, possible rotational axis directions of P are represented by $\boldsymbol{x}$ which satisfies the following inequality:

$$[(\boldsymbol{v}-\boldsymbol{v}_c)\times\boldsymbol{f}]^T\boldsymbol{x}\geq 0$$

where $\times$, $\boldsymbol{v}$ and $\boldsymbol{v}_c$ denote the outer product of vectors, the position vector of p, and the foot of the perpendicular from $\boldsymbol{v}$ to the rotational axis, respectively. $(\boldsymbol{v}-\boldsymbol{v}_c)$ is normalized.

When p and q are vertices, the method proposed by Hirukawa et al.[5] is used. In their method, constraint inequalities can be derived depending on the angles of two faces adjacent to p and q. If all angles of any two faces adjacent to p and q are less than π, $\boldsymbol{f}$ corresponds to the faces adjacent to p and q and separating the neighborhood of P and Q. Motion with $\boldsymbol{x}$ which satisfies any one constraint of them is possible. These constraints can be represented by logical-or connections. For example, in Fig. 4, P can be translated along $\boldsymbol{x}$ which satisfies the inequalities in which $\boldsymbol{f}$ is either $\boldsymbol{f}_1$ or $\boldsymbol{f}_2$.

Otherwise if there are two faces whose angles are greater than π, $\boldsymbol{x}$ must satisfy both of two constraints in which $\boldsymbol{f}$ corresponds to either of the two faces. These constraints can be represented by logical-and connections.

For other contact states between polyhedra, we can represent the constraints using the method for face-vertex and vertex-vertex contact cases. For instance, when p and q are both faces, constraints can be represented by face-vertex cases where vertices are those of the convex hull of the contact region.

Therefore, by only considering the constraints with logical-and connections, we can represent the constraints as follows:

$$\boldsymbol{F}\boldsymbol{x}\geq\boldsymbol{0},\quad \boldsymbol{F}\in\boldsymbol{R}^{n\times 3}, \tag{1}$$

where n denotes the number of constraints.

If P has a cylindrical contact, it can be translated only along the cylindrical axis. This constraint can be represented as follows:

$$|\boldsymbol{a}^T\boldsymbol{x}|=1,$$

where $\boldsymbol{a}$ is a normalized vector of the cylindrical axis. P can also be allowed to rotate about only the cylindrical axis. This restriction can be represented as follows:

$$|\boldsymbol{a}^T\boldsymbol{x}|=1,\quad |\boldsymbol{a}^T(\boldsymbol{v}_x-\boldsymbol{v}_a)|=1,$$

where $\boldsymbol{v}_x$ and $\boldsymbol{v}_a$ denote points on the rotational axis and the cylindrical axis, respectively. $(\boldsymbol{v}_x-\boldsymbol{v}_a)$ is normalized and $\boldsymbol{v}_x\neq\boldsymbol{v}_a$. Therefore translational and rotational motions which satisfy the following condition are allowed:

$$\boldsymbol{x}=\pm\boldsymbol{a}. \tag{2}$$

To obtain several motions which maintain contact states, we impose the following constraint.

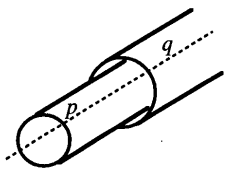

Fig. 2 Cylindrical contact

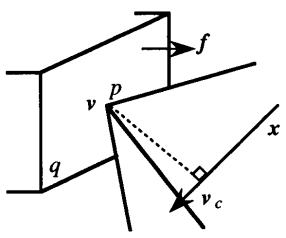

Fig. 3 Contact state and possible motion

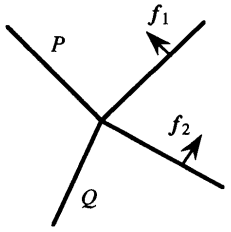

Fig. 4 Constraint from vertex-vertex contact

$$|\boldsymbol{x}| \leq x_{\text{norm}}, \quad (3)$$

where x_{norm} denotes the Euclidean norm. Though this constraint is nonlinear, it can be approximated by linear inequalities. The solutions satisfying the expressions (1)-(3) becomes the region from a polyhedral convex set. The extreme points of it can be calculated. Some of them, may not maintain the contact states as shown in Fig. 5.

The rotational motion of a part is restricted to that about the axis of a contacted cylindrical surface, that is, if no cylindrical contact exists, a verified part has to be disassembled without rotational motion.

3.3 Calculation of configuration to transit contact states

The transition of contact states can be classified as follows:

(1) The sum of dimensionality of contacting elements decreases, where dimensionalities of a vertex, an edge, and a surface are 0, 1, and 2, respectively. If any element and its bounds are in contact, only the maximum value of the sum of dimensionality is considered. For instance, in Fig. 6, an edge of P and a face of Q, a vertex of P and a face of Q, an edge of P and an edge of Q are in contact at once. In this case, the edge-face contact has the maximum value of the sum of dimensionality, $1+2=3$, and is used. If P is translated along the direction $\boldsymbol{x}$ as shown in Fig. 6, the edge-face contact turns into the vertex-edge contact whose sum of dimensionality is 1. Thus the sum of dimensionality decreases. We call this configuration where contact state transition occurs contact decreasing configuration (CDC).

(2) Contact states are changed by the collision between parts P and Q. We call such a configuration contact increasing configuration (CIC).

When a cylindrical contact exists P and rotates about its axis, a case that contact states do not change occurs. This rotational motion is not required for removing a part, and can be excluded.

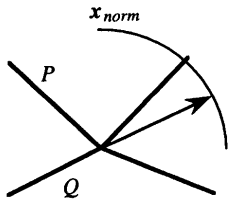

Fig. 5 Translation without keeping contact state

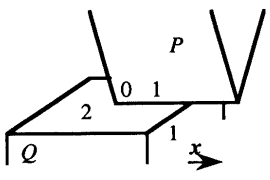

Fig. 6 Contact states and dimensionalities

3.3.1 Calculation of contact decreasing configuration If p and q are both faces and the motion is a translational one, the CDC $\boldsymbol{w}_d$ is that shown in either Fig. 7(a) or 7 (b). For the case in Fig. 7(a), we can determine the CDC by calculating the intersections between straight lines along translational direction $\boldsymbol{x}$ and edges bounding q. We can determine the CDC for the case in Fig. 7(b) in a similar manner.

If the motion is a rotational one, the CDC is that shown in Fig. 7(c) or (d), and we can find the CDC by calculating the intersections between circles and straight lines.

For the other contact states between polyhedra, the CDC can also be classified into four types as shown in Fig. 7. For example, when p is a vertex, q is a face and the motion is a translational one, we can use the case in Fig. 7(a).

For the cylindrical contact, the CDC exists for only translational motion from the assumption in Section 3. 1, and can readily be calculated.

3.3.2 Calculation of contact increasing configuration The CIC $\boldsymbol{w}_d$ is obtained by detecting the collision between P and Q. Several methods have been developed for the collision detection[7],[8]; however, in this research, the translational and rotational motions are dealt with separately, and the collisions can be detected in an analytical manner. This enables maintenance of the consistency of contact states, even though the part moves repeatedly.

Collisions between polyhedra can be classified as follows:

(1) Vertex of P and face of Q
(2) Face of P and vertex of P
(3) Edge of P and edge of Q

For case(1), when the motion is a translational one, $\boldsymbol{w}_d$ is obtained by calculating the intersections between straight lines whose initial points are vertices

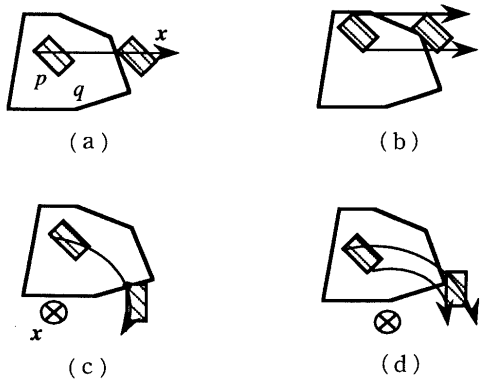

Fig. 7 Contact decreasing configuration

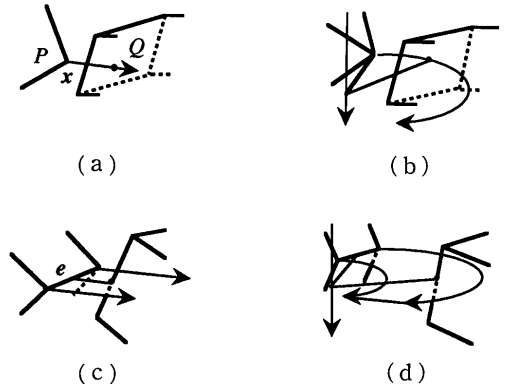

Fig. 8 Contact increasing configuration

of P along the translational direction and faces of Q as shown in Fig. 8 (a). When the motion is a rotational one, $\boldsymbol{w}_d$ is determined by calculating the intersections between circles created by vertices of P and a rotational axis, and faces of Q, as shown in Fig. 8 (b). This procedure can easily be extended to case (2).

For case (3), when the motion is a translational one, we can determine $\boldsymbol{w}_d$ by calculating the intersections between swept surfaces of edges of P and edges of Q as shown in Fig. 8 (c). When the motion is a rotational one, the swept surfaces are classified by calculating the inner product of an edge $\boldsymbol{e}$ of P and the rotational axis $\boldsymbol{x}$. If $\boldsymbol{e}^T\boldsymbol{x}=0$, the swept surface is a planar surface. If $|\boldsymbol{e}^T\boldsymbol{x}|=1$, it becomes a cylindrical surface. If $0<\boldsymbol{e}^T\boldsymbol{x}<1$ and straight lines along $\boldsymbol{e}$ and $\boldsymbol{x}$ do not intersect, it is a conical surface. Otherwise, it is represented by a hyperbolical surface of one sheet. The CIC is determined by calculating the intersections of these swept surfaces and edges of Q as shown in Fig. 8 (d).

3. 4 Algorithm for verifying disassemblability

Figure 9 shows the algorithm for verifying the disassemblability. First we calculate $\boldsymbol{X}_s$ which denotes a set of possible motions $\boldsymbol{x}_{sk}$ $(k=1, \cdots, K)$ calculated from any configuration $\boldsymbol{w}_s$. Then, for each k, we find $\boldsymbol{w}_{sk}^d$ and $\boldsymbol{w}_{sk}^i$, meaning the CDC and CIC calculated with $\boldsymbol{x}_{sk}$ and $\boldsymbol{w}_s$, and set the nearer configuration of them to $\boldsymbol{w}_s$ to $\boldsymbol{w}_{sk}$. Replacing $\boldsymbol{w}_s$ by $\boldsymbol{w}_{sk}$ and repeating this procedure, a tree structure is constructed whose nodes and arcs denote configurations and possible motions of a verified part, respectively.

When $\boldsymbol{w}_{sk}^i$ does not exist for an arbitrary translational motion, we decide that a part is disassemblable. An upper boundary $N_{\max}$ for the allowable number of times that motion direction for which a part can be changed is introduced. If a state, in which a part is disassemblable, cannot be found even though all nodes satisfying the upper boundary are checked,

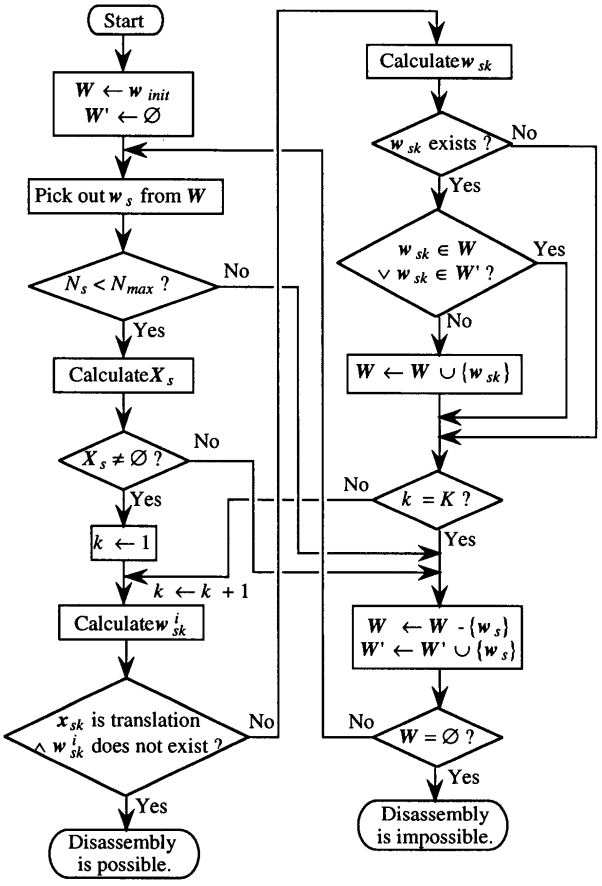

Fig. 9 Procedure flow chart

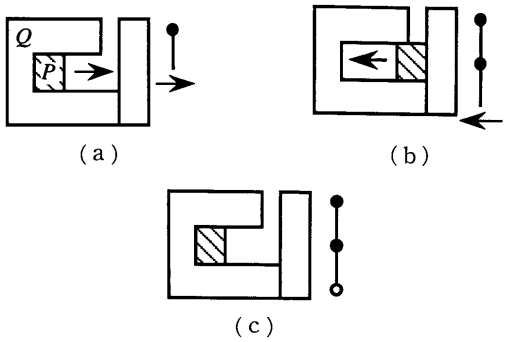

Fig. 10 Obtaining same configuration

the part is not disassemblable. If a configuration which is exactly the same as that which has already been entered is found, it is not entered, since possible motions calculated from them are the same. In this procedure, the case that a verified part moves along the same path repeatedly does not occur. For instance, in Figs. 10 (a) and 10 (c), a verified part has the same configuration, so that state (c) is not entered. The current implementation is based on the breadth-first search.

In Fig. 9, $\boldsymbol{w}_{\text{init}}$, N_s, $\boldsymbol{W}$ and $\boldsymbol{W}_s$ denote an initial configuration of a verified part, the number of times

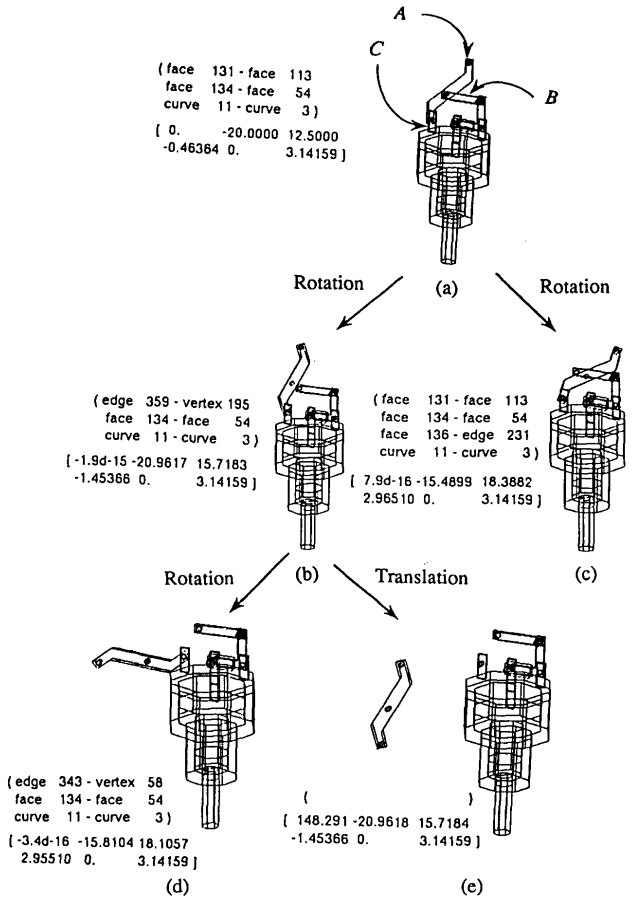

Fig. 11 Execution result

that motion direction changes before reaching the configuration w_s, a set of unchecked nodes, and a set of checked nodes, respectively.

4. Example

We applied our developed system to a practical mechanical part of a breaking booster shown in Fig. 11, denoted by A. We set N_{max} as 3. Figure 11(a) shows a state in which two contact states exist: one is between faces of parts A and B whose normals are directed upward and downward from the surface of this paper, respectively. The other is a cylindrical contact between parts A and C. The states in Figs. 11(b) and 11(c) are calculated from the state in Fig. 11(a). No CDC exists for the translational motion calculated from the state in Fig. 11(b). Figure 11(e) shows a disassembled state. This calculation has been done within 10 minutes using a workstation with CPU MC68020.

5. Conclusion

We proposed a method to verify the assemblability of mechanical products on the assumption that assembly operation can be replaced by disassembly operation.

We present the methods to calculate the possible motions and to detect the configurations where contact state transition occurs.

Based on them, we replace the disassemblability verification method with the disassembly path generation using tree structural data whose nodes and arcs denote configurations and possible motions of a verified part, respectively.

A software system is implemented and its efficiency is shown by applying it to a practical mechanical product.

The proposed method cannot verify the assemblability rigorously; however, we believe it has high practical utility for design verification and assembly planning.

References

(1) Kitajima, K., Nishiyama, T. and Yoshikawa, H., Development of an Interactive Machine Assemblability Check System Based on Oriented Connectivity Graph of Machine, J. JSPE, (in Japanese), Vol. 49, No. 2 (1983), p. 208.

(2) Woo, T. C. and Dutta, D., Automatic Disassembly and Total Ordering in Three Dimensions, Trans. ASME, J. for Ind., Vol. 113, No. 1 (1991), p. 207.

(3) Lozano-Perez, T., A Configuration Space Approach, IEEE Trans. Comput., Vol. 32, No. 2 (1983), p. 108.

(4) Loeff, L. W. and Soni, H., An Algorithm for Computer Guidance of a Manipulator in between Obstacles, Trans. ASME, J. for Ind., Vol. 97, No. 3 (1975), p. 836.

(5) Hirukawa, H., Matsui, T. and Takase, K., Automatic Generation Algorithm of Possible Motion from Geometric Model under the Constraint by Contact between Polyhedra, Prep. RSJ, (in Japanese), (1989), p. 525.

(6) Hirai, S. Analysis and Planning of Manipulation Using the Theory of Polyhedral Convex Cones, Dr. Dissertation, Kyoto Univ., (1990), p. 9.

(7) Mizugaki, Y., Kimura, F. and Sata, T, A Simplified Method for Collision Detection Based on Geometric Modelling, J. JSPE, (in Japanese), Vol. 50, No. 6 (1984), p. 944.

(8) Shimada, K., A Method of Detecting Collisions in Geometric Simulation, J. JSPE, (in Japanese), Vol. 54, No. 8 (1988), p. 162.