9-14-2003

Efficient Contact State Graph Generation for Assembly Applications

Feng Pan
Marquette University

Joseph M. Schimmels
Marquette University, joseph.schimmels@marquette.edu

IEEE Transactions on Robotics and Automation, (2003): 2592-2598. DOI.

Efficient Contact State Graph Generation for Assembly Applications

Feng Pan
Department of Mechanical Engineering, Marquette University, Milwaukee, WI
J.M. Schimmels
Department of Mechanical Engineering, Marquette University, Milwaukee, WI

Abstract:

An important aspect in the design of many automated assembly strategies is the ability to automatically generate the set of contact states that may occur during an assembly task. In this paper, we present an efficient means of constructing the set of all geometrically feasible contact states that may occur within a bounded set of misalignments (bounds determined by robot inaccuracy). This set is stored as a graph, referred to as an Assembly Contact State Graph (ACSG), which indicates neighbor relationships between feasible states. An ACSG is constructed without user intervention in two stages. In the first stage, all hypothetical primitive principle contacts (PPCs; all contact states allowing 5 degrees of freedom) are evaluated for geometric feasibility with respect to part-imposed and robot-imposed restrictions on relative positioning (evaluated using optimization). In the second stage, the feasibility of each of the various combinations of PPCs is efficiently evaluated, first using topological existence and uniqueness criteria, then using part-imposed and robot-imposed geometric criteria.

SECTION I.

Introduction

The ability to automatically generate a description of contact configuration space is an important component in the development of automated assembly strategies. In fine motion planning [1], [2], the contact configuration space (C-obstacle) must be explicitly specified. This type of description is very difficult to obtain for planar rigid body motion [3] and an unresolved problem for general spatial motion [4]. As a result, more recent automated assembly strategies [5], [6] use high-level descriptions of contact configuration space.

Here, we describe an automated means of obtaining a high-level description of the various contact topologies that may occur during the assembly of two polyhedral objects. The topological description obtained contains all geometrically feasible contact states stored as a connected graph that identifies the neighbor relationships among the contact states. Geometric feasibility is assessed using both part-geometry imposed and robot-positioning imposed constraints on the relative positioning of the two objects.

Others have addressed means of obtaining high-level descriptions of contact configuration space. Xiao et al. have addressed the automatic generation of a high- level description of contact state space for both planar [7] and spatial applications [8]. Their approach is based on relaxing topological constraints on user-defined highly-constrained seed contact states, then testing the feasibility of each less constrained state by evaluating multiple available (kinematically unconstrained) motions. The connected sub-graph obtained from each seed is then merged with all other subgraphs to obtain the complete contact state graph containing neighbor relationships.

Goeree et al. have also addressed the issue of automatically generating a description of high-level contact state space for both planar [9] and spatial applications [10]. Their approach is based on evaluating all possible combinations of different single-point contacts. The geometric feasibility of each hypothesized contact state is evaluated using numerical optimization. Although user-defined seed contact states are not required in their approach, the number of possible combinations considered is extremely large, even for the simplest polyhedra.

In this paper, we describe a hybrid approach that combines a method of describing contact states and their neighbor relationships similar to that of [8] with a method of testing the feasibility of contact states similar to that of [10]. This hybrid approach will allow: 1) the set of contact states that may occur to be reliably generated without user intervention, 2) contact state neighbor information to be stored as a graph, and 3) realistic bounds on relative positioning uncertainties to be incorporated as constraints when evaluating feasible contact states. These user defined bounds on translational and rotational misalignment (determined by the positioning capabilities of the robot arm) will be used to limit the contact states considered to those that may realistically occur.

An outline of the paper is provided below. Section II reviews some existing terminology used to describe contact topology and also defines some additional terms used in this paper. Section III provides a description of the Assembly Contact State Graph (ACSG) generation approach. In section IV, our method of evaluating geometrical feasibility is identified. In section V, means of reducing the number of contact states using conditions associated with topological existence and uniqueness are presented. In section VI, a simple spatial example demonstrating the reduction in contact states resulting from topological and geometrical restrictions is presented. A brief summary and discussion is presented in section VII.

SECTION II.

Notation

In this section, brief reviews of the notations used in this paper to describe the topology of polyhedral objects and the topology of contact are presented.

A. Object Description

A polyhedral object can be described by the set of *faces*, *edges* and *vertices* known as the object's *surface elements* (SE). Here, a surface element is denoted by *seoi*, where *o* indicates the object to which the element belongs and *i* indicates the element number. We define the following subsets of surface elements:

$v_{cv}^{o} \equiv v_i^o | v_i^o$is a convex vertex of object O};

$v_{cc}^{o} \equiv v_i^o | v_i^o$ is a concave vertex of object o};

$e_{cv}^{o} \equiv e_i^o | e_i^o$ is a convex edge of object o};

$e_{cc}^{o} \equiv e_i^o | e_i^o$ is a concave edge of object o};

$f^{o} \equiv f_i^o | f_i^o$ is a face of object o};

We also define $\Gamma(\cdot)$ asan operator that obtains the boundary set of a SE. The boundary set of an edge is its two boundary vertices, and the boundary set of a face contains its boundary edges as well as its boundary vertices.

A face element may contain multiple interior polygons. To uniquely identify the boundary elements of each polygon, we define $\Gamma_k(f_i^o)(k \geq slant0)$ as an operator that obtains the set of boundary elements belonging to the *kth* polygon. $\Gamma_0(f_i^o)$ returns the set associated with the outer polygon. $\Gamma_k(f_i^o)$ returns the set associated with inner polygon k (if it exists).

B. Contact State Description

As defined in [8], a *Principle Contact* (PC) describes contact between any two surface elements. A single PC is denoted as $PC_i = (se_p^A - se_q^B)$, where se_p and se_q are surface elements of objects A and B, respectively. There are ten topologically and kinematically distinct types of PCs [8] categorized into: 1) degeneratePCs[1], and 2) non-degenerate PCs.

A PC can also be described in terms of its neighbor relationships. One characterization is based on *Less Constrained Neighbors* (LCNs) [8] of a PC. By definition, $PC_i = (se_p^A - se_q^B)$ is a LCN of $PC_j = (se_r^A - se_s^B)$, iff one of the following holds:

- $se_P^A \in \Gamma(se_r^A)$; and $se_q^B \in \Gamma(se_s^B)$;
- $se_p^A \in \Gamma(se_r^A)$ and $se_q = se_s$;
- $se_p = se_{rs}$ and $se_q^B \in \Gamma(se_s^B)$.

With this convention, the least constrained PC is (v-v) contact.

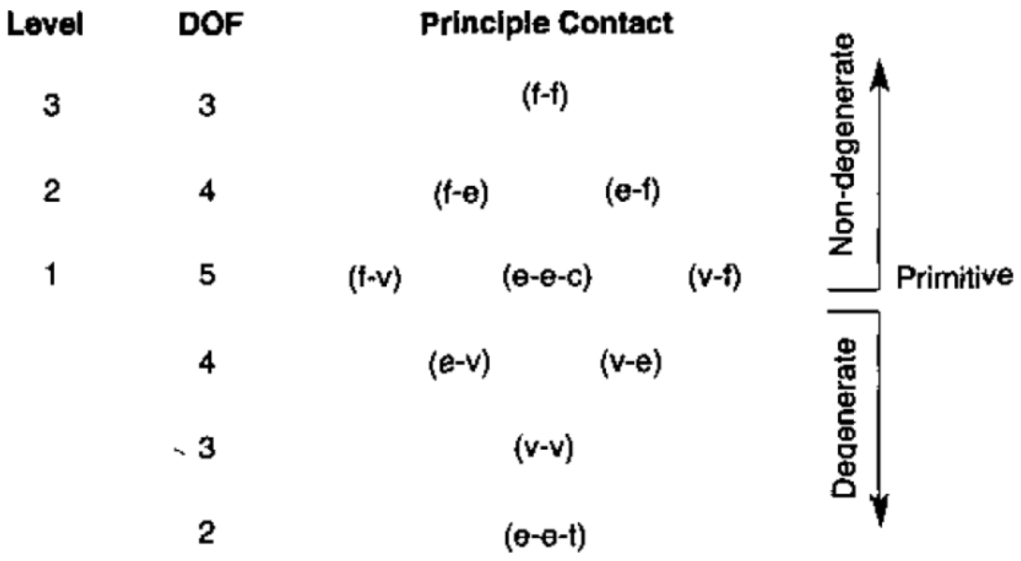


Fig. 1. Classification of PCs by Level of Constraint.

Another means of classifying PCs is by the amount of kinematic constraint imposed on an object's motion (to maintain contact of that type). Some PCs allow 5 degrees-of-freedom (DOF); whereas, another allows only 2. The kinematically least constrained PCs are those corresponding to single-point contact (($\mathrm{f} - \mathrm{v}$), ($\mathrm{e} - \mathrm{e}$ −c), and (v-f) contact; “primitive contacts” in [10]). We refer to these types of contact as *Primitive Principle Contacts (PPCs*).

The set of all PPCs, *PPC*, for two contacting objects is given by:

$$PPC \equiv \{PC_i | DOF(PC_i) = 5, \forall PC_i\} \quad (1)$$

where *DOF* (·) is an operator that obtains the DOF of a PC. The size of the set is given by:

$$\begin{aligned} card(PPC) \quad &= card(f^{A}) \times card(v_{cv}^{B}) \\ &+card(e_{cv}^{A}) \times card(e_{cv}^{B}) \quad (2) \\ &+card(v_{cv}^{A}) \times card(f^{B}) \end{aligned}$$

where *card* (·) returns the cardinality of a set.

PPCs provide the lowest level of kinematic constraint. The remaining non-degenerate PCs can alternately be described by combinations of PPCs. Their PPC descriptions, however, are non-unique. Both PC and PPC descriptions are used in this work. We refer to a description involving multiple PPCs *as* a "low level" description; and one based on other nondegenerate PCs as a "high level" description. A classification of PCs based on level of constraint is presented in Fig. 1.

A *Contact State* (CS) describes the topology of more general contact. Here, we use this term to describe a set containing one or more PCs, e.g., $CS_i = \{PC_j, j = 1,2 \cdots n\}$. "Higher-level" CSs are associated with a larger number of PPCs used in the "low level" description.

Related to the level of contact description is an operator that yields the lowest level description of a CS. The *Kinematically Least Constrained* (KLC) operator obtains the PPCs associated with each of the higher level PCs in a CS. The set associated with a given CS is given by:

$$PPC_{CS_i} \equiv \{KLC(CS_i)\} \subset PPC \quad (3)$$

An *Assembly Contact State Graph* (ACSG) identifies the set of all geometrically feasible CSs (Section III) and the neighbor relationships among them. In this paper, we describe the algorithm used to find the set of valid CSs (denoted by ACSG) given by:

$$\mathrm{ACSG} \equiv \{CS_i | CS_i \mathrm{isfeasible}, \forall CS_i\} \quad (4)$$

without explicitly defining the data structures used to identify neighbor relationships.

SECTION III.

Contact State Generation Approach

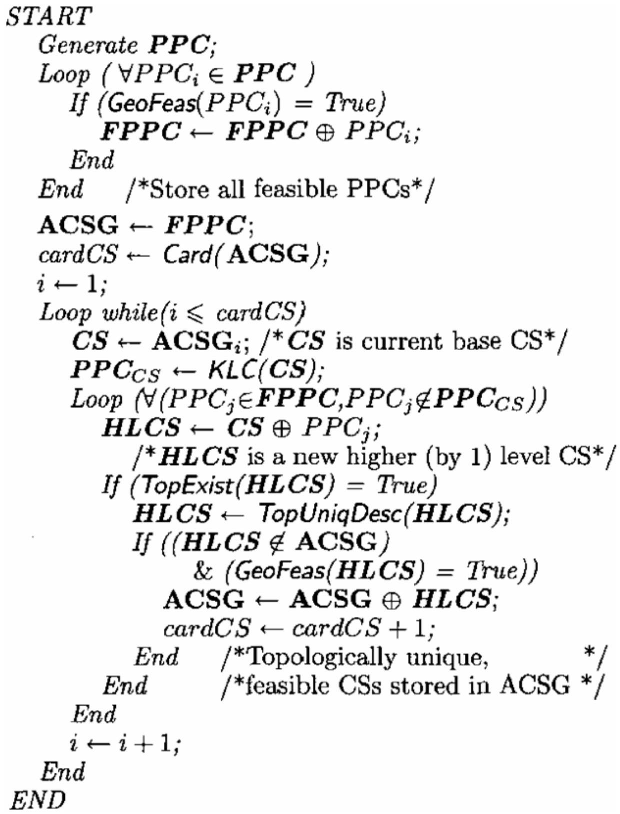
The set of contact states that may occur during the assembly of two polyhedral objects is described below. Our approach is similar to [10] in that combinations of single-point contact primitives are assessed for geometric feasibility using optimization. The approach *is* different in that the set of combinations of contact primitives evaluated using optimization is significantly reduced in number [2].

To reduce the number of optimization procedures performed, we consider the generation of the ACSG in two stages. In the first stage, all geometrically infeasible contact primitives are eliminated prior to considering their combinations. In the second stage, the logic of the algorithm prevents all higher-level PPC combinations having infeasible subsets from even being generated. This stage also eliminates all topologically impossible contact states. Also, because unique higher-level descriptions (PC combinations) are stored in the ACSG, all redundant descriptions of high-level contact are eliminated from geometrical evalution.

Our approach is similar to that of [8] in that PCs are used to describe contact primitives and neighbor relationships among contact states are used. Unlike [8], however, seed highly-constrained configurations are not required as user input. All required geometric input is extracted from computer models of the parts to be assembled (from IGES files) with the only user input being the bounds on the reasonable relative positions of the parts (determined by robot pose uncer-tainty). Also, because optimization is used, the evaluation of geometric feasibility considers general spatial motion (not only a finite set of motions) and concurrently considers both part-imposed and robot-imposed restrictions on part relative positioning.

In general terms, the entire ACSG is obtained from a set of geometrically feasible PPCs. Only topologically distinct combinations of PPCs are considered. Feasible PPCs combinations are assessed with respect to topological existence and topological uniqueness prior to using optimization. The procedure terminates when no new contact state can be generated. A pseudo-code algorithm description is given below.III.

ACSG Generation Algorithm



```
START
  Generate $\boldsymbol{PPC}$;
  Loop ( $\forall PPC_i \in \boldsymbol{PPC}$ )
    If (GeoFeas($PPC_i$) = True)
      $\boldsymbol{FPPC} \leftarrow \boldsymbol{FPPC} \oplus PPC_i$;
    End
  End     /*Store all feasible PPCs*/
  $\mathbf{ACSG} \leftarrow \boldsymbol{FPPC}$;
  $cardCS \leftarrow$ Card($\mathbf{ACSG}$);
  $i \leftarrow 1$;
  Loop while($i \leqslant cardCS$)
    $\boldsymbol{CS} \leftarrow \mathbf{ACSG}_i$; /*$\boldsymbol{CS}$ is current base CS*/
    $\boldsymbol{PPC}_{CS} \leftarrow$ KLC($\boldsymbol{CS}$);
    Loop ($\forall (PPC_j \in \boldsymbol{FPPC}, PPC_j \notin \boldsymbol{PPC}_{CS})$)
      $\boldsymbol{HLCS} \leftarrow \boldsymbol{CS} \oplus PPC_j$;
        /*$\boldsymbol{HLCS}$ is a new higher (by 1) level CS*/
      If (TopExist($\boldsymbol{HLCS}$) = True)
        $\boldsymbol{HLCS} \leftarrow$ TopUniqDesc($\boldsymbol{HLCS}$);
        If (($\boldsymbol{HLCS} \notin \mathbf{ACSG}$)
              & (GeoFeas($\boldsymbol{HLCS}$) = True))
          $\mathbf{ACSG} \leftarrow \mathbf{ACSG} \oplus \boldsymbol{HLCS}$;
          $cardCS \leftarrow cardCS + 1$;
        End     /*Topologically unique,          */
      End       /*feasible CSs stored in ACSG */
    End
    $i \leftarrow i + 1$;
  End
END
```

The operator *GeoFeas* (·) determines the geometrical feasibility of a CS (Section IV); the operator *TopExist* (·) checks topological existence (Section V.A); *Top-UniqDesc* (·) maps a CS to its highest level description (Section V.B), *KLC* (·) obtains all PPCs of a CS (Section II), and ⊕ indicates Minkowski set addition.

SECTION IV.

Geometrical Evaluation

In our approach the geometrical feasibility of every PPC and every topologically feasible CS is evaluated. In this section, both part-imposed and robot-imposed restrictions on part relative positioning are addressed using optimization.

A. Part-Imposed Criteria

In [10], conditions describing contact for (f-v), (e-e-c), and (v-f) have been defined. In the following sub-sections, we define similar conditions for the remaining PCs, i.e., for (f-e), (e-f) and (f-f) contact. First, because some results from [10] are useful in evaluating the remaining PCs, they are briefly reviewed below.

The Euclidean distances of a vertex from a face for (f-v) and (v-f) contact are given by:

$$\begin{aligned} h_1^{fv} &= d_S^E(f_i^A, v_l^B) = (\vec{r}_{v_l^B} - \vec{r}_{f_i^A})^T \vec{n}_{f_i^A} = 0 \\ h_1^{vf} &= d_S^E(v_l^A, f_i^B) = (\vec{r}_{v_l^A} - \vec{r}_{f_i^B})^T \vec{n}_{f_i^B} = 0 \end{aligned} \qquad (5)(6)$$

where $\vec{r}_{v_l^o}$ and $\vec{r}f_i^o$ denote positions of the vertex and an arbitrary point on the face, respectively, and $\vec{n}_{f_i^o}$ denotes the face normal.

If face f_c^o is convex, it can be represented by the intersection of its supporting plane and the negative half-spaces of $\hat{a}$set of bounding planes, $bp(f_c^o)$. If face f_i^o is non-convex, it can be decomposed into a set of convex faces $cv(f_i^o)$. To ensure that a vertex contacts a face, the following conditions for (f-v) and (v-f) contact, respectively, must be satisfied.

$$
\begin{aligned}
h_2^{fv}(f_i^A, f_l^B) &= \prod_{f_c^A \in cv(f_i^A)} \sum_{f_k^A \in bp(f_c^A)} \mu(-d_s^E(f_k^A, v_l^B)) = 0 \\
h_2^{vf}(v_l^A, f_i^B) &= \prod_{f_c^B \in cv(f_i^B)} \sum_{f_k^B \in bp(f_c^B)} \mu(-d_s^E(v_l^A, f_k^B)) = 0
\end{aligned}
$$

(7)(8)

where

$$\mu(x) = \begin{cases} x^2 & x < 0 \\ 0 & x \geq slant0 \end{cases} \quad (9)$$

If two edges intersect, they satisfy:

$$h_1^{ee}(e_p^A, e_q^B) = (\vec{v}_{e_p^A} \times \vec{v}_{e_q^B})^T (\vec{r}_{v_i^A} \times \vec{r}_{v_m^B}) = 0 \quad (10)$$

and

$$
\begin{aligned}
h_2^{ee}(e_p^A, e_q^B) \quad &= \mu(\alpha^A_{e_p^A e_q^B}) + \mu(l_{e_p^A} - \alpha^A_{e_p^A e_q^B}) \\
&+ \mu(\alpha^B_{e_p^A e_q^B}) + \mu(l_{e_q^B} - \alpha^B_{e_p^A e_q^B}) = 0
\end{aligned}
\quad (11)
$$

where

$$
\begin{aligned}
\alpha^A_{e_p^A e_q^B} &= \frac{1}{1-c^2_{e_p^A e_q^B}} [\vec{u}_{e_p^A} - c_{e_p^A e_q^B} \vec{u}_{e_q^B}]^T \vec{r}_{v_l^B, v_m^A}, \\
\alpha^B_{e_p A e_q^B} &= \frac{1}{1-c^2_{e_p^A e_q^B}} [c_{e_p^A e_q^A} \vec{u}_{e_q^A} - \vec{u}_{e_q^B}]^T, \vec{r}_{v_i^B v_{\mathrm{m}}^A}, \quad (12)(13)(14) \\
c_{e_p^A e_q^B} &= (\vec{u}_{e_p^A})^T \vec{u}_{e_q^B},
\end{aligned}
$$

and l_e denotes the length of an edge, $\vec{u}_{e_p^o}$ denotes a vector along an edge, and $\vec{r}_{v_1^B, v_m^A}$ is a vector from $v_m^A \in \Gamma(e_p^A) \mathrm{to} v_l^B \in \Gamma(e_q^B)$.

A.1 Face-Edge Contact $(f_i^A - e_p^B)$

For a face to contact an edge: 1) the edge must be in the supporting plane of the face, and either 2a) a boundary vertex of the edge lies in the face or 2b) the edge contacts a boundary edge of the face.

For condition 1, if an edge is in a plane, both boundary vertices of the edge lie in the plane. This is satisfied if both of the following are satisfied:

$$\begin{aligned} h_1^{fe}(f_i^A, e_p^B) &= d_s^E(f_i^A, v_l^B) = 0 \\ h_2^{fe}(f_i^A, e_p^B) &= d_s^E(f_i^A, v_m^B) = 0 \end{aligned} \quad (15)(16)$$

where $\{v_l^B, v_m^B = \Gamma(e_p^B)$.

For condition 2a, if a boundary vertex of an edge lies in the face, $h_2^{fv}(f_i^A, v_l^B)$ or $h_2^{fv}(f_i^A, v_m^B)$ is zero. Equivalenty, $\Pi h_2^{fv}(f_i^A, v_p^B)(\forall v_p^B \in \Gamma(e_p^B))$ is zero.

For condition 2b, if the edge contacts a boundary edge of the face, then $\exists e_q^A \in \Gamma(f_i^A)$ such that $h_2^{ee}(e_q^A, e_p^B)$ is zero. Equivalently, $\prod h_2^{ee}(e_q^A, e_p^B)(\forall e_q^A \in \Gamma(f_i^A))$ is zero.

The entire second condition can be expressed as:

$$\begin{aligned} h_3^{fe}(f_i^A, f_p^B) \quad &= \prod_{v_n^B \in \Gamma(e_p^B)} h_2^{fv}(f_i^A, v_n^B) \\ &\cdot \prod_{e_q^A \in \Gamma(f_i^A)} h_2^{ee}(e_q^A, e_p^B) = 0 \end{aligned}$$

(17)

A.2 Edge-Face Contact $(e_p^A - f_i^B)$

Similarly, the conditions for Edge- Face contact are:

$$\begin{aligned} h_1^{ef}(e_p^A, f_i^B) &= d_s^E(v_l^A, f_i^B) = 0 \\ h_2^{\mathrm{ef}}(e_p^A, f_i^B) &= d_s^E(v_m^A, f_i^B) = 0 \end{aligned} \quad (18)(19)$$

where $\{v_l^A, v_m^A\} = \Gamma(e_p^A)$ and

$$h_3^{ef}(e_p^A, f_i^B) \quad = \prod_{v_n^A \in \Gamma(e_p^A)} h_2^{vf}(v_n^A, f_i^B) h_2^{vf}(v_m^A, f_j^B) \cdot \prod_{e_q^B \in \Gamma(f_i^B)} h_2^{ee}(e_p^A, e_q^B) = 0 \qquad (20)$$

A.3 Face-Face Contact $(f_i^A - f_j^B)$

For two faces to be in contact, 1) the two faces must be co-planar, and either 2a) a vertex of one face lies in the other, or 2b) two boundary edges of each face intersect.

For condition 1, coplanar faces satisfy:

$$\begin{aligned} h_1^{ff} &= |\vec{n}_{f_i^A} \times \vec{n}_{f_j^B}| = 0 \\ h_2^{ff} &= (\vec{r}_{f_j^B} - \vec{r}_{f_i^A})^T \vec{n}_{f_i^A} = 0 \end{aligned} \qquad (21)(22)$$

where $\vec{n}_{f_i^o}$ is a face normal and $\vec{r} f_i^o$ is arbitrary location on the face.

For condition 2a, if a vertex of f_j^B lies on f_i^A, then $\exists v_l^B \in \Gamma(f_j^B)$ such that $h_2^{fv}(f_i^A, v_l^B)$ is zero. Equivalently, $\Pi h_2^{vf}(f_i^A, v_l^B)(\forall v_l^A \in \Gamma(f_i^B))$ is zero.

Similarly, if a vertex of f_i^A lies on f_j^B, $\Pi h_2^{vf}(v_m^A, f_i^B)(\forall v_m^A \in \Gamma(f_i^A))$ is zero.

For condition 2b, if two boundary edges of each face intersect, then $\exists e_p^A \in \Gamma(f_i^A)$ and $\exists e_q^B \in \Gamma(f_j^B)$ such that $h_2^{ee}(e_p^A, e_q^B)$ is zero. Equivalently,

$$\prod\prod h_2^{ee}(e_p^A, e_q^B)(e_p^A \in \Gamma(f_i^A), e_q^B \in \Gamma(f_j^B)) \text{is zero.}$$

The entire second condition can be expressed as:

$$h_3^{ff}(f_i^A f_j^B) \quad = \prod_{v_l^B \in \Gamma(f_j^B)} h_2^{fv}(f_i^A, v_l^B) \prod_{v_m^A \in \Gamma(f_j^A)} h_2^{vf}(v_m^A, f_j^B)$$

$$\cdot \prod_{e_p^A \in \Gamma(f_i^A)} \prod_{e_q^B \in \Gamma(f_j^B)} h-_2^{ee}(e_p^A, e_q^B) = 0$$

(23)

A.4 Geometrical Conflict

In addition to satisfying the geometric conditions that define a contact state, additional geometric conditions preventing object penetration must also be satisfied. One measure of polyhedral object penetration is known as the *penetration growth distance* [11]. Like [10], we use this measure to assess geometric conflict.

B. Robot-Imposed Criteria

In generating an ACSG, only those contact states that may realistically occur during an assembly task are considered. Relative positions are required to be within the set of reasonable misalignments determined by robot pose uncertainty. These conditions can be expressed in the form:

$$\vec{X}_{min} \leq slant\vec{X} \leq slant\vec{X}_{max} \quad (24)$$

where $\vec{X} \in \mathbb{R}^6$ denotes the relative misalignment of the bodies (3 translational and 3 orientational variables) and $\vec{X}_{min} \in \mathbb{R}^6$ and $\vec{X}_{max} \in \mathbb{R}^6$ denote the lower and upper bounds of the misalignment.

C. Geometrical *Evaluation* Summary

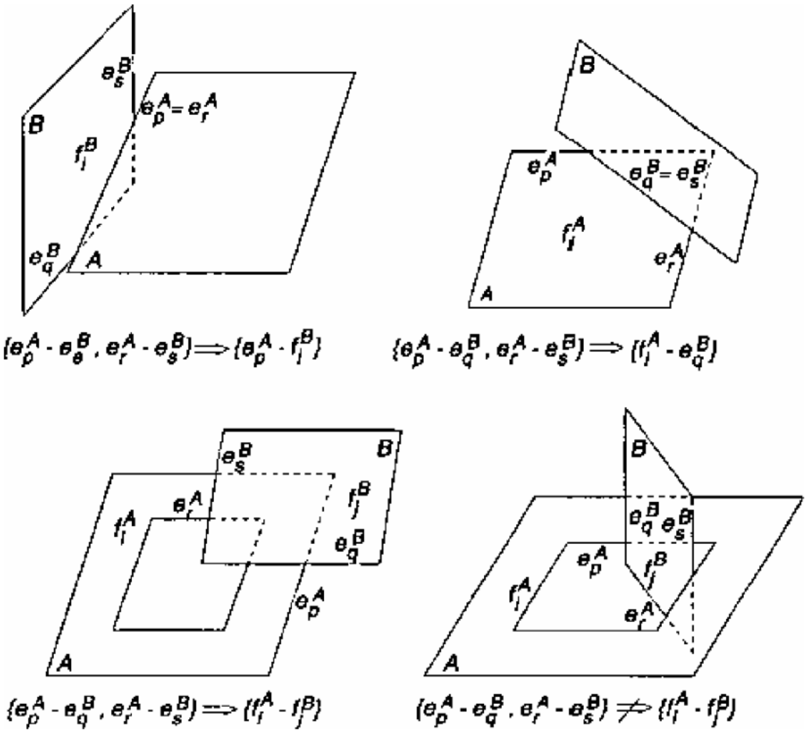
All geometric conditions are evaluated using optimization. All part-imposed and robot-imposed geometric constraints are contained in the optimization objective function. The penetration growth distance [11] and robot-imposed constraints (Section IV.B) are considered in all evaluations. The remaining conditions depend on the specific type of contact state being evaluated (Section IV.A).

The relative positioning of the parts is optimized so that contact between the specified topological features is achieved within the positioning uncertainty bounds of the robot and without causing object penetration. If the optimization returns an objective function value of zero, the contact state is geometrically feasible.

SECTION V.

Topological Evaluation

Because geometrical evaluation requires time-consuming optimization procedures, we wish to minimize the number of optimization procedures required when evaluating all combinations of PPCs. Therefore, prior to using optimization to determine the geometrical feasibility of non-PPC CSs, the CS is first evaluated to determine whether is satisfies topological existence and uniqueness criteria. Because topological evaluation can be accomplished with logic, it can be accomplished relatively easily and quickly.



B
e_s^B
$e_p^A = e_r^A$
f_l^B
e_q^B
A
$\{e_p^A - e_e^B, e_r^A - e_s^B\} \Longrightarrow \{e_p^A - f_l^B\}$
B
e_p^A
$e_q^B = e_s^B$
f_l^A
e_r^A
A
$\{e_p^A - e_q^B, e_r^A - e_s^B\} \Longrightarrow \{f_l^A - e_q^B\}$
e_s^B
B
e_r^A
f_j^B
f_l^A
e_q^B
e_p^A
A
$\{e_p^A - e_q^B, e_r^A - e_s^B\} \Longrightarrow \{f_l^A - f_j^B\}$
B
e_q^B e_s^B
e_p^A
f_j^B
f_l^A
e_r^A
A
$\{e_p^A - e_q^B, e_r^A - e_s^B\} \not\Longrightarrow \{f_l^A - f_j^B\}$

Fig. 2. Example PPC Combinations Mapped to a Higher-Level. Different types of ((e-e),(e-e)) combinations are mapped to the appropriate higher-level descriptions.

A. Topological Existence

Some combinations of PPCs cannot exist. For example, a vertex cannot contact two non-adjacent faces simultaneously. If a CS contains the following combinations of PPCs. $CS_i = \{\cdots, v_l^A - f_i^B, v_l^A - f_i^B, \cdots\}$ where $i \neq j$ and no exists$e_p^B \in e_{cc}^B$ s.t. $\{e_p^B\} - \in (\Gamma(f_i^B) \cap \Gamma(f_j^B))$, then *CSi* is topologically infeasible.

Similarly, if a CS contains the following PPCs: $CS_j = \{\cdots f_i^A - v_l^B, f_i^A - v_l^B, \cdots\}$ where *i≠j* and noexists$e_p^A \in e_{cc}^A$such that$\{e_p^A\} \in (\Gamma(f_i^A) \cap \Gamma(f_j^A))$, then CSj is topologically infeasible.

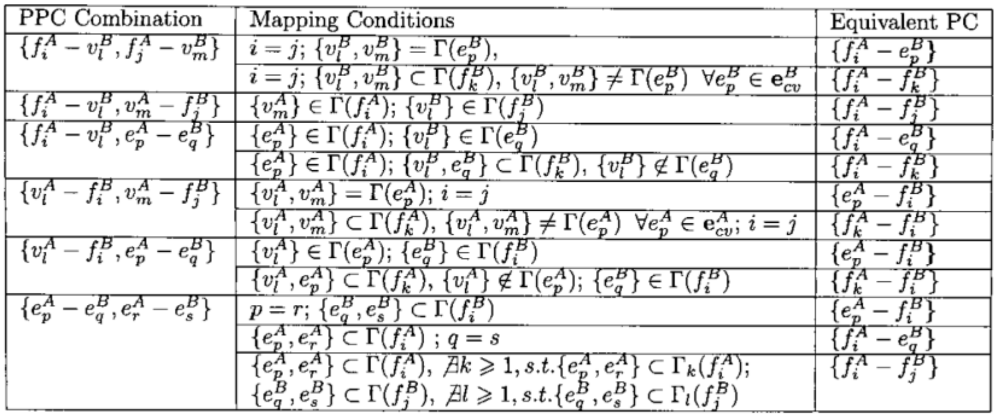
B. Topological Uniqueness

As stated in Section II, “low level” (PPC) descriptions of contact states are not unique. Redundant descriptions of higher-level contact states can be eliminated from further consideration using a set of topological evaluations to obtain higher-level descriptions of combinations of lower level PCs. Once a specified CS is identified as being feasible, its highest level description is stored in the ACSG. As a result, subsequently generated redundant descriptions of this CS are not evaluated using optimization. In this subsection, the mapping relationships needed to obtain a higher-level description are identified.

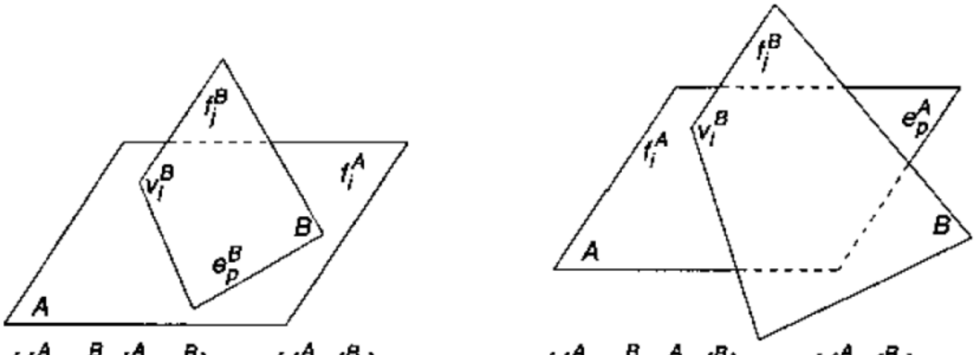
B.1 Combination of Two PPCs

Each PPC in a CS provides one additional level of constraint to the object motion. Mapping two PPCs to an equivalent PC normally leads to a second level PC. In some cases, however, because of the specific topological relationship of the surface elements, this mapping leads to a third level PC. The specific mapping relationships having this characteristic are identified in Table 1. Figure 2 illustrates examples for which (e-e),(e-e)) PPC combinations are mapped to a higher level description.

Table I Mapping Relationships for Combinations OP Two PPCs



PPC Combination	Mapping Conditions	Equivalent PC
$\{f_i^A - v_l^B, f_j^A - v_m^B\}$	$i = j;\ \{v_l^B, v_m^B\} = \Gamma(e_p^B),$	$\{f_i^A - e_p^B\}$
	$i = j;\ \{v_l^B, v_m^B\} \subset \Gamma(f_k^B),\ \{v_l^B, v_m^B\} \neq \Gamma(e_p^B)\ \ \forall e_p^B \in \mathbf{e}_{cv}^B$	$\{f_i^A - f_k^B\}$
$\{f_i^A - v_l^B, v_m^A - f_j^B\}$	$\{v_m^A\} \in \Gamma(f_i^A);\ \{v_l^B\} \in \Gamma(f_j^B)$	$\{f_i^A - f_j^B\}$
$\{f_i^A - v_l^B, e_p^A - e_q^B\}$	$\{e_p^A\} \in \Gamma(f_i^A);\ \{v_l^B\} \in \Gamma(e_q^B)$	$\{f_i^A - e_q^B\}$
	$\{e_p^A\} \in \Gamma(f_i^A);\ \{v_l^B, e_q^B\} \subset \Gamma(f_k^B),\ \{v_l^B\} \notin \Gamma(e_q^B)$	$\{f_i^A - f_k^B\}$
$\{v_l^A - f_i^B, v_m^A - f_j^B\}$	$\{v_l^A, v_m^A\} = \Gamma(e_p^A);\ i = j$	$\{e_p^A - f_i^B\}$
	$\{v_l^A, v_m^A\} \subset \Gamma(f_k^A),\ \{v_l^A, v_m^A\} \neq \Gamma(e_p^A)\ \ \forall e_p^A \in \mathbf{e}_{cv}^A;\ i = j$	$\{f_k^A - f_i^B\}$
$\{v_l^A - f_i^B, e_p^A - e_q^B\}$	$\{v_l^A\} \in \Gamma(e_p^A);\ \{e_q^B\} \in \Gamma(f_i^B)$	$\{e_p^A - f_i^B\}$
	$\{v_l^A, e_p^A\} \subset \Gamma(f_k^A),\ \{v_l^A\} \notin \Gamma(e_p^A);\ \{e_q^B\} \in \Gamma(f_i^B)$	$\{f_k^A - f_i^B\}$
$\{e_p^A - e_q^B, e_r^A - e_s^B\}$	$p = r;\ \{e_q^B, e_s^B\} \subset \Gamma(f_i^B)$	$\{e_p^A - f_i^B\}$
	$\{e_p^A, e_r^A\} \subset \Gamma(f_i^A)\ ;\ q = s$	$\{f_i^A - e_q^B\}$
	$\{e_p^A, e_r^A\} \subset \Gamma(f_i^A),\ \nexists k \geqslant 1, s.t. \{e_p^A, e_r^A\} \subset \Gamma_k(f_i^A);$ $\{e_q^B, e_s^B\} \subset \Gamma(f_j^B),\ \nexists l \geqslant 1, s.t. \{e_q^B, e_s^B\} \subset \Gamma_l(f_j^B)$	$\{f_i^A - f_j^B\}$



$\{f_i^A - v_l^B, f_i^A - e_p^B\} \Longrightarrow \{f_i^A - f_j^B\}$ $\{f_i^A - v_l^B, e_p^A - f_j^B\} \Longrightarrow \{f_i^A - f_j^B\}$

Fig. 3. Example PC Combinations Mapped to a Higher Level. Two types of ((f-v(f-e)) and ((f−v),(e−f)) contact states are mapped to the appropriate higher level description.

B.2 Combination of a PPC and a Second Level PC

Some combinations of a PPC with a second level PC can yield contact states that are uniquely described by a higher level PC. The combinations and the associated relationships having this characteristic are identified in Table 2. Figure 3 illustrates examples for which ((f-v),(f-e)) and ((f-v),(e-f)) are mapped into a higher level description.

SECTION VI.

Example

Figure 4 illustrates two parts to be assembled. Proper assembly is achieved when the coordinate frames illustrated are coincident.

The number of PPCs associated with the object is calculated using Eq. (2). For this case, we obtain:

$$cars(PPC) = 8 \cdot 8 + 17 \cdot 12 + 10 \cdot 6 = 328$$

If all possible combinations of PPCs were considered, the number of optimization procedures required would be $2^{PPC} = 2^{328} \approx 5.5(10)^{98}$, a number too large to evaluate in practice.

In our approach, the set of PPCs is first evaluated for feasibility prior to evaluating their combinations. For misalignment bounds given by:

$$\begin{aligned} \vec{X}_{min} &= [-5,-5,0,-\pi/10 - \pi/10,-\pi/10]^T \\ \vec{X}_{max} &= [5,5,240,\pi/10,\pi/10,\pi/10]^T \end{aligned}$$

the number of PPCs considered in subsequent evaluation (the number of feasible PPCs) is reduced to 13. These feasible PPCs are:

$$\begin{aligned} FPPC = \quad & \{f_0^A - v_0^A, f_0^A - v_1^B, f_0^A - v_2^B, f_0^A - v_3^B, \\ & f_1^A - v_0^B, f_1^A - v_3^B, f_2^A - v_0^B, f_2^A - v_3^B, \\ & e_0^A - e_0^B, e_1^A - e_2^B, e_0^A - e_0^B, e_0^A - e_4^B, \\ & e_0^A - e_7^B\} \end{aligned}$$

where ***FPPC*** indicates the set of feasible PPCs. Therefore for this example, the number of optimization procedures conducted in the first stage is 328.

In the second stage, the ACSG is generated from combinations of these 13 FPPCs. If all possible combinations of FPPCs were considered for this case, 213=8192 combinations would be considered.

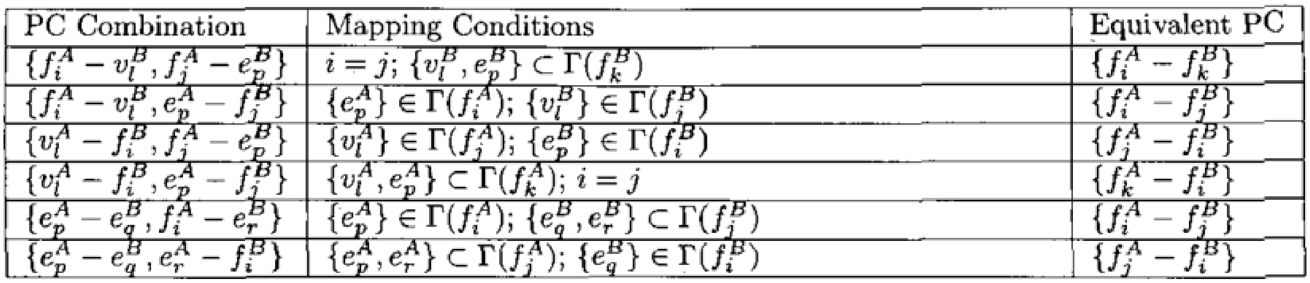
However, using our approach, not all combinations are generated (because they involve infeasible subsets) and not all combinations are considered using optimization (only 424).

Therefore, the total number of optimization procedures for this case is 752 (328 + 424), corresponding to a reduction of approximately 96 orders of magnitude.

For this example, an ACSG containing 126 feasible contact states was generated.

Subsequent analysis of the ACSG $\{v_l^A, u_m^A\} = \Gamma(e_p^A); i = j$indicated that some generated contact states were redundant. These cases corresponded to situations where higher level descriptions would better describe a feasible configuration. For example $CS_A =< f_1^A - f_4^B, f_0^A - e_1^B >$ was obtained based on the procedure introduced in this paper. This contact state, however, should be described by $CS_B =< f_1^A - f_4^B, f_0^A - f_0^B >$, since CS_A can only occur in a configuration associated with CS_B.

Table II Mapping Relationships of Combination of Two PCs



PC Combination	Mapping Conditions	Equivalent PC
$\{f_i^A - v_l^B, f_j^A - e_p^B\}$	$i = j; \{v_l^B, e_p^B\} \subset \Gamma(f_k^B)$	$\{f_i^A - f_k^B\}$
$\{f_i^A - v_l^B, e_p^A - f_j^B\}$	$\{e_p^A\} \in \Gamma(f_i^A); \{v_l^B\} \in \Gamma(f_j^B)$	$\{f_i^A - f_j^B\}$
$\{v_l^A - f_i^B, f_j^A - e_p^B\}$	$\{v_l^A\} \in \Gamma(f_j^A); \{e_p^B\} \in \Gamma(f_i^B)$	$\{f_j^A - f_i^B\}$
$\{v_l^A - f_i^B, e_p^A - f_j^B\}$	$\{v_l^A, e_p^A\} \subset \Gamma(f_k^A); i = j$	$\{f_k^A - f_i^B\}$
$\{e_p^A - e_q^B, f_i^A - e_r^B\}$	$\{e_p^A\} \in \Gamma(f_i^A); \{e_q^B, e_r^B\} \subset \Gamma(f_j^B)$	$\{f_i^A - f_j^B\}$
$\{e_p^A - e_q^B, e_r^A - f_i^B\}$	$\{e_p^A, e_r^A\} \subset \Gamma(f_j^A); \{e_q^B\} \in \Gamma(f_i^B)$	$\{f_j^A - f_i^B\}$

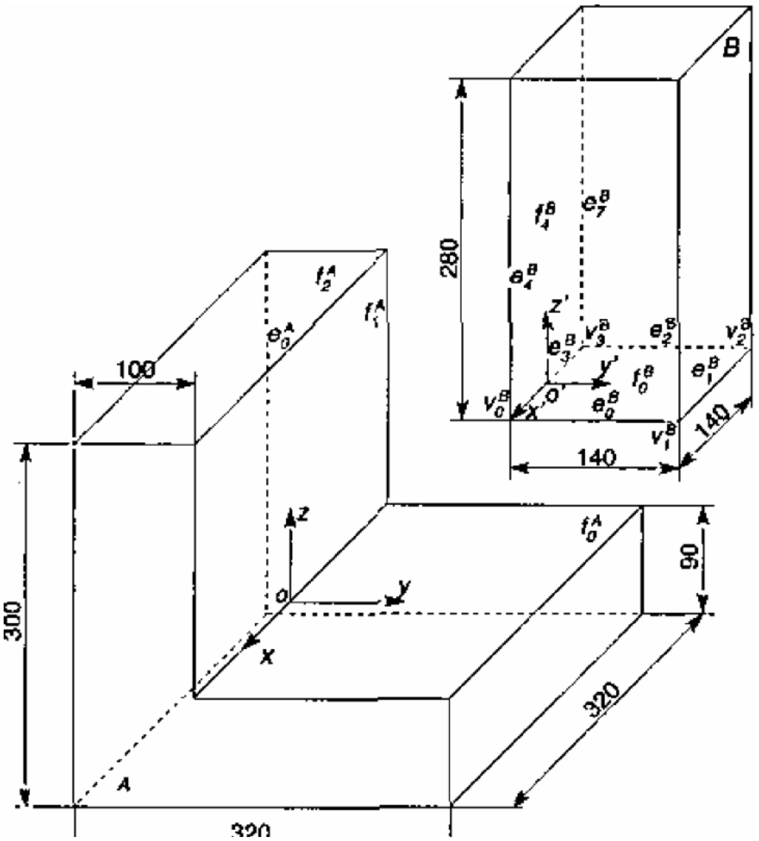


Fig. 4. Example Assembly. Proper assembly is achieved when the coordinate frames are coincident. To avoid clutter, only those surface elements that can be contacted during assembly are labeled.

SECTION VII.

Summary and Discussion

In this paper, we have presented means of generating a connected graph containing the set of contact states that may occur during an assembly operation. The approach does not require user-specified seed contact states and generates only the “useful” portion of the graph - only those contact states that could realistically occur.

In this paper, we only discussed object topological information. The contact states generated by this procedure include some redundant descriptions due to the specific geometry of two objects. The elimination of these redundant contact states using geometrical in- formation without optimization is being addressed in ongoing work.

ACKNOWLEDGMENT

This research is supported by the National Science Foundation under grant IIS 0010017 and by a Ford Motor Company URP award.

References

[1] T. Lozano-Perez, M. T. Mason, and R. H. Taylor, "Automatic synthesis of fine-motion strategies for robots," International Journal of Robotics Research, vol. 3, no. 1, 1984.

[2] Buckley, S. J., "Planning compliant motion strategies," The International Journal of Robotics Research, vol. 8, no. 5, 1989.

[3] Brost, R. C., "Computing metric and topological properties of configuration-space obstacles," in IEEE International Conference on Robotics and Automation, 1989, pp. 170-176.

[4] Jean-Claude Latombe, Robot Motion Planning, Kluwer Academic Publishers, Boston, 1991.

[5] B.J. McCarragher and H. Asada, "The discrete event control of robotic assembly tasks," ASME Journal of Dynamic Systems, Measurement and Control, vol. 117, pp. 384-393, September 1995.

[6] S. Huang and J. M. Schimmels, "Sufficient conditions used in admittance selection for planar force-guided assembly," in Proceedings of the IEEE International Conference on Robotics and Automation, Washington, D.C., May 2002, 538-543.

[7] J. Xiao and L. Zhang, "Contact constraint analysis and determination of geometrically valid contact formations from possible contact primitives," IEEE Transactions on Robotics and Automation, vol. 13, no. 3, pp. 456-466, 1997.

[8] J. Xiao and X. Ji, "Automatic generation of high-level contact space," The International Journal of Robotics Research, vol. 20, no. 7, pp. 584-606, 2001.

[9] B. B. Goeree, E. D. Fasse, and M. M. Marefat, "Verifying contact hypotheses of planar, polyhedral objects using penetration growth distance," Robotics and Computer-Integrated Manufacturing, vol. 17, no. 3, pp. 233-246, 2001.

[10] B. B. Goeree, E. D. Fasse, and M. M. Marefat, "Determining feasible contact state of pairs of spatial polyhedra," in Proceeding of the IEEE International Conference on Robotics and Automation, San Francisco, CA, 2000, pp. 1396-1401.

[11] C. Ong and E. G. Gilbert, "Growth distances: New measures for object separation and penetration," IEEE Transactions on Robotics and Automation, vol. 12, no. 6, pp. 888-903, 1996.