4-26-2004

Robust Procedures for Obtaining Assembly Contact State Extremal Configurations

Pan Feng
Marquette University

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

Accepted version. Published as a part of the *Proceedings. ICRA '04. 2004 IEEE International Conference on Robotics and Automation, 2004*. DOI.

Robust Procedures for Obtaining Assembly Contact State Extremal Configurations

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

Keywords

Robustness, Assembly, Admittance, Genetic algorithms, Personal communication networks, USA Councils, Data structures, Industrial engineering

Abstract:

Two important components in the selection of an admittance that facilitates force-guided assembly are the identification of: 1) the set of feasible contact states, and 2) the set of configurations that span each contact state, i.e., the extremal configurations. We present a procedure to automatically generate both sets from CAD models of the assembly parts. In the

procedure, all possible combinations of principle contacts are considered when generating hypothesized contact states. The feasibility of each is then evaluated in a genetic algorithm based optimization procedure. The maximum and minimum value of each of the 6 configuration variables spanning each contact state are obtained by again using genetic algorithms. Together, the genetic algorithm approach, the hierarchical data structure containing the states, the relationships among the states, and the extremals within each state are used to provide a reliable means of identifying all feasible contact states and their associated extremal configurations.

SECTION I.

INTRODUCTION

In the development of automated assembly strategies, the types of part misalignment that may occur must be known. The subset of configuration space (C-space) that corresponds to contact (the C-space obstacle), however, is very difficult to describe even for planar cases [1]. As a consequence, recent research efforts [2], [3], [4], [5] have been directed toward identifying a set of topological descriptions of contact for polygonal and polyhedral parts. These “high-level” topological descriptions are referred to as contact states.

In addition to contact states, the sets of configurations that bound translational and rotational misalignment in each contact state also play a key role in the design of the manipulator admittance that achieves force-guided assembly [6]. An admittance designed for force guidance maps the contact force associated with a type of misalignment into a motion that reduces that misalignment. The configuration “extremals” represent configurations that have extreme values in misalignment information within the specified contact state. As described in [6], by imposing conditions on the admittance at a finite number of representative “extremal” configurations, force-guided assembly is ensured for all of the infinite number of intermediate configurations.

A. Related Work

Previously, Xiao and Ji [4] systematically described contact states in terms of *Principle Contacts* (PCs). A PC describes contact between any two surface elements (i.e., faces, edges, or vertices) and is denoted as $PC_i = \{se_p^A - se_q^B\}$, where se_j^o identifies surface element j of object o. There are 6 distinct types of non-degenerate PCs (face-vertex, edge-edge-cross, vertex-face, face-edge, edge-face, and face-face). A contact state indicates the set of PCs that are simultaneously attained and is denoted by $\mathbf{CS}_i = \{PC_j, j = 1,2 \cdots, n\}$. Using their PC-based description of the contact states, Xiao and Ji [4] developed the ability to identify the contact states that may occur for 3D polyhedra in spatial motion. Contact state feasibility is evaluated in their approach using a direct search of the configuration space obstacle. Their approach requires the user to provide one or more “seed” contact states (locally most constrained contact states) which can be very difficult to obtain for some cases.

Goeree et al. [5] described contact states in terms of combinations of *Primitive Contacts* which are of 3 types (face-vertex, vertex-face, and edge-edge). This type of contact description, however, yields non-unique descriptions of some contact states. They developed more reliable means of assessing the feasibility of hypothesized contact states using traditional numerical optimization procedures. Their procedure for generating the set of hypothesized contact states, however, considers all possible combinations of primitive contacts yielding an extremely large set, even for the simplest polyhedra.

Pan and Schimmels [7] introduced an algorithm to generate the set of contact states for an assembly operation, known as Assembly Contact State Graph (ACSG). The ACSG is systematically generated by first evaluating the least constrained contact states. By combining a single point contact state (PPC) with a feasible contact state, a new higher level (more constrained) hypothesized contact state is generated. The procedure has the following benefits: 1) no user supplied “seed” information is required, and 2) realistic limits on possible part misalignment can be imposed.

In previous work directed toward identifying a range of configurations within a contact state, Xiao and Zhang [8] computed the angular distance between two contacting polyhedra for a given rotational axis passing through the point of contact. The rotational axis was determined by the contact state. In [9], Ji and Xiao presented an approach to planning motions compliant to contact states. A component of the approach involved determining configurations within the range of translation or rotation for a specified contact state. The approach introduced in [8] was used to determine the rotational range. For translation, rather than explicitly determining the range, a configuration within the range was determined only.

B. Approach

To reliably generate contact states and associated extremal configurations, optimization is used to search the configuration space. The search is facilitated by using a data structure (associated with the assembly contact state graph (ACSG) [7]) that both records the contact states and stores the configuration extremals. When checking the feasibility of a more constrained contact state (contact states at higher levels in the ACSG), the extremal configurations from lower levels are used to restrict the range of the search. As such, the constraint relationships contained in the ACSG are used to improve procedure robustness both in checking the feasibility and in obtaining extremals of contact states.

The procedure of identifying an *assembly* contact state (as introduced in [7]) and the approach to obtaining associated extremal configurations take into account the range of configurations determined by robot inaccuracy. Here, *an assembly* contact state is feasible if there is a representative configuration with the specified configuration range.

C. Overview

This paper describes means of reliably generating the set of assembly contact states and identifying the extremal configurations within each contact state using optimization. Section II reviews the procedure for obtaining the set of feasible assembly contact states from standard geometrical descriptions (DXF files) of polyhedral parts using a nested optimization. Section III introduces the methods for obtaining the extremal configurations within a feasible contact state. A strategy for improving robustness using extremal configurations and relationships among the contact states is presented in Section IV. Section V presents a simple spatial example demonstrating the algorithm. A discussion and brief summary are presented in Section VI.

SECTION II.

Contact State Feasibility

A reliable means of obtaining all assembly contact states without user intervention was introduced in [7]. The general algorithm has the following components:

1. Generate the set of PPCs from DXF files of the parts (A PPC is a PC having 5 degrees of freedom (face-vertex, vertex-face and edge-edge-cross));
2. Combine feasible PPCs to generate a hypothesized contact state;
3. Map the hypothesized contact state into the highest level topological description;
4. Check the feasibility of the hypothesized contact state;
5. Repeat Steps 2–4 until no additional PPC combinations can be generated.

In this section, the conditions used to evaluate assembly contact state feasibility (item 4) are briefly reviewed. These conditions are based on three considerations: 1) the surface elements of the hypothesized contact state must be in contact (the CS Feature Criteria); 2) there must be no geometric conflict between the parts (the Non-Penetration Criteria); and 3) the configuration must be within the user specified bounds of misalignment (the Bounded-Misalignment Criteria).

A. CS Feature Criteria

In [5], Goeree *et al.* defined a set of mathematical conditions for vertex-face, face-vertex and edge-edge-cross PCs. In [7], similar conditions were defined for face-edge, edge-face and face-face PCs. Each of these conditions consists of two types of sub-conditions, *Intersection Conditions* and *Bounding Conditions*.

1) Intersection Conditions

Each PC has at least one intersection condition. Intersection conditions require that the supporting geometrical element (plane for face, line for edge) must intersect a feature from the other part (*e.g.*, feature is coplanar or collinear). Intersection Conditions that are satisfied have the following form [7]:

$$h_{PC}^{\mathbf{1}}(\mathbf{T}) = 0 \quad (1)$$

where the subscript *PC* indicates one of six types of nondegenerate PCs. The independent variable **T** of this condition is the relative configuration of two objects, containing 6 configuration variables.

2) Bounding Conditions

Each PC has multiple bounding conditions. The bounding conditions require that the intersection occurs within some bounds (*e.g.*, point contact within bounded face, line intersection within line segment). Bounding Conditions that are satisfied have the following form [7]:

$$h_{PC}^{2}(\mathbf{T}) = 0$$

3) Combination Conditions

By combining the Intersection Conditions with the Bounding Conditions, the CS Feature Criteria is satisfied if the following holds:

$$h_{PC}(\mathbf{T}) = W_1 h_{PC}^{1}(\mathbf{T}) + W_2 h_{PC}^{2}(\mathbf{T}) = 0 \quad (3)$$

where W_1 and W_2 are the scaling factors for a PC. Their values are determined by the dimension of two objects and the level of PC (amount of constraint).

The conditions for all single PC contact states have this form. For a CS consisting of multiple PCs, $\mathrm{CS}_i = \{PC_j, j = 1,2, \cdots n\}$. The condition for every PC_j must be satisfied simultaneously, indicated by:

$$\sum_{PC_j \in \mathbf{CS}_i} h_{PC_j}(\mathbf{T}) = 0 \quad (4)$$

Note that the left side of (4) must be non-negative [7]. Therefore, if (4) is satisfied for a specific configuration T, contact of type $\mathbf{CS}_i$ can occur.

B. Non-Penetration Criteria

In addition to requiring contact of the specific surface elements, penetration between two parts at other locations must be prevented. Part penetration can be assessed using a measure of polyhedral object penetration known as the *growth distance* [10].

The growth distance is a function of configuration, denoted as:

$$d_p^G(\mathbf{T}) = d_p^G(A, B) \geq 0. \quad (5)$$

If there is penetration between two objects, the growth distance $d_p^G(A, B)$ is positive. The growth distance is zero when contact is obtained at any location of the body.

As described in [10], the calculation of growth distance for a given configuration of polyhedral objects involves constrained linear optimization.

Equation (5) together with (4) define the conditions used to evaluate whether a given configuration corresponds to a feasible configuration within a specified contact state.

C. Bounded-Misalignment Criteria

Some generally feasible configurations cannot occur in an assembly operation due to the restrictions on possible part misalignment.

The configuration of an unconstrained rigid body T can be described by 6 independent variables **t**, consisting of three translation variables **p** and three rotation variables θ [11].

$$\mathbf{t} = [\mathbf{p}\theta] \quad (6)$$

Limits associated with robot or fixture inaccuracy can be expressed as:

$$t_{i_{min}} \leqslant t_i \leqslant t_{i_{max}}, \quad (7)$$

where $t_{i_{min}}$ and $t_{i_{max}}, (i = 1,2, \cdots ,6)$ indicate the lower and upper bounds, respectively, of the misalignment for each of the 6 variables in (6).

D. Configuration Search Algorithm

The procedures above provide means of checking whether a configuration corresponds to a specific feasible assembly contact state. A procedure for finding a feasible configuration, if one exists, is needed. In this section, an optimization procedure used to reliably identify a feasible configuration for each contact state is described.

Because the criteria described above ((4) – (5)) involve nonlinear relationships, a nonlinear optimization procedure is required. The design variables in the optimization are the configuration variables in **t**. If all the conditions are satisfied ((4), (5) and (6)) for some configuration, this hypothesized contact state is assembly-feasible.

Problems of this type can be solved using standard *gradient search* [5] methods. These methods, however, have the following limitations:

- Gradient search methods start from a single initial value, and the optimal result depends highly on the initial value chosen. Because of the nonlinear conditions, there exists many spurious local minima. Therefore, it is difficult to obtain the global minimum for the optimization procedure even for the simplest contact state (face-vertex, vertex-face or edge-edge-cross).
- Gradient search methods require gradients, which are difficult to obtain due to the nested optimization used here (growth distance is obtained by optimization).

A genetic algorithm is a relatively new class of stochastic nonlinear optimization that does not have these limitations. Genetic algorithms start with a large number of randomly generated initial values (referred to as the population). Then using reproduction, crossover and mutation [1], new generations of improved (higher fitness) individuals are obtained [12]. The general optimization format of a genetic algorithm is to obtain the minimum value of a objective function subject to bounds on the design variables.

In our optimization, (7) is used to upper and lower bound each of the variables in **t**. If a configuration is feasible, both CS Feature Criteria and Non-Penetration Criteria are satisfied, i.e. both $d_p^G(\mathbf{T})$ and $\sum h_{PC_j}(\mathrm{T})$ are zero. So, to evaluate a hypothesized contact state $\mathbf{CS}_i$, these two conditions are combined to obtain the objective function:

$$f(\mathbf{T}) = \sum_{PC_j \in \mathbf{CS}_i} h_{PC_j}(\mathbf{T}) + W_3 d_p^G(\mathbf{T}) \tag{8}$$

where $h_{PC_i}(\mathbf{T})$ is the appropriate CS feature criteria for PC_j (4), and W_3 is a scaling factor, whose value is determined by the dimensions of two objects.

The feasibility of a contact state is evaluated by optimizing the objective function (8) subject to the bounded misalignment conditions (7). If the objective value is zero, this indicates that the Non-Penetration Criteria and CS Feature Criteria are both satisfied and that there exists a feasible configuration within the range considered. As such, the hypothesized contact state is assembly-feasible. If the optimal value of (8) is greater than zero, this indicates that no configuration obtained by the genetic algorithm corresponds to a feasible configuration for the hypothesized contact state and the hypothesized contact state is assembly-infeasible.

E. Algorithm Implementation

As stated previously, the procedure of checking the feasibility of a hypothesized contact state involves nested optimization. The inner optimization calculates the growth distance for a given configuration (as described in Section II.B) obtained using the simplex method. The outer optimization searches for a feasible configuration in a nonlinear optimization and is obtained using a genetic algorithm (as described in Section II.D).

SECTION III.

Contact State Extremals

Extremal configurations represent configurations that have extreme values in misalignment within a specific contact state. Formally,

III.

Definition 1

An extremal configuration of a contact state is a special configuration having at least one of its six configuration variables at its maximum or minimum value. In this section, we present an algorithm to identify the extremal configurations for a contact state.

A. Obtaining Extremal Configurations

Because an extremal configuration must be a configuration within the specified contact state, it must satisfy: i) the CS Feature Criteria, (4); ii) the Non-Penetration Criteria (5); and iii) the Bounded-Misalignment Criteria, (7). The problem of finding the lower bounded extremal configurations for independent variable *tk* can be formulated as a constrained optimization problem:

$$min t_k s.t. \begin{cases} d_p^G(\mathbf{T}) = 0 \\ \sum h_{PC_j}(\mathbf{T}) = 0 \\ t_{i_{min}} \leq t_i \leq t_{i_{min}} \forall i = 1,2,\cdots,6 \end{cases} \quad (9)$$

Similarly, the optimization to determine the upper bounded extremals of each independent variable t_k is given by

$$min - t_k \text{s.t.} \begin{cases} d_p^G(\mathbf{T}) = 0 \\ \sum h_{PC_j}(\mathbf{T}) = 0 \\ t_{i_{min}} \le t_i \le t_{i_{max}} \forall i = 1,2,\cdots,6 \end{cases} \quad (10)$$

To solve this type of constrained optimization problem using genetic algorithms, a penalty method is used to incorporate some constraint equations into the objective function. The 2 new objective functions are expressed as:

$$min f_1(\mathbf{T}) = t_k + \lambda_1 (\sum_{PC_j \in \mathbf{CS}_i} h_{PC_j}(\mathbf{T}) + W_3 d_p^G(\mathbf{T}))$$
$$min f_2(\mathbf{T}) = -t_k + \lambda_2 (\sum_{PC_j \in \mathbf{CS}_i} h_{PC_j}(\mathbf{T}) + W_3 d_p^G(\mathbf{T}))$$
(11)(12)

where λ_1 and λ_2 are large positive numbers known as penalty factors, and W_3 is a scaling factor defined in (8).

The objective functions (11) (12) are similar to (8) in that all are subject to limits on the design (configuration) variables. A genetic algorithm is again used to obtain configuration extremals for each assembly contact state.

SECTION IV

Robustness Improvement

The conditions for evaluating the feasibility of more constrained contact states are larger in number and more nonlinear than the conditions for less constrained contact states. This occurs because each higher level CS requires additional CS Feature Criteria (4). The robustness of most optimization routines decreases when the number of constraints and the nonlinearity of the constraints increase. In this section, a robustness improvement strategy based on a search space refinement that uses the relationships among contact states and extremal configurations (both stored in the ACSG data structure) is described.

A. Contact State Relations

In [4], Xiao and Ji identified the neighbor relations of contact states based on the topology of the two polyhedral parts. Here, we use a restrictive form of that information.

For two **PCs**, $\mathbf{PCs}, PC_a = (se_i^A - se_j^B), PC_b = (se_k^A - se_l^B)$, if: 1) se_i^A is se_k^A and se_j^B is the boundary of se_l^B, or 2) se_i^A is the boundary of se_k^A and se_j^B is se_l^B, then PC_a is the *close less constrained neighbor* (CLCN) PC of PC_b and PC_b is the *close more constrained neighbor* (CMCN) PC of PC_a.

The relationships of PCs are consistent with the classification of PCs by the level of constraint in [7]. The CLCN PC of a PC is a PC in the next lower level[2], and the CMCN PC is in the next higher level.

For two CSs, $\mathbf{CS}_a = \{PC_i\}$ and $\mathbf{CS}_b = \{PC_j\}, \mathbf{CS}_a$ is the *close less constrained neighbor* (CLCN) CS of $\mathbf{CS}_b$, and $\mathbf{CS}_b$ is the *close more constrained neighbor* (CMCN) CS of $\mathbf{CS}_a$, if one of the following two conditions is satisfied:

- if the number of PCs in $\mathbf{CS}_a$ is equal to that in $\mathbf{CS}_b$, and there are a pair of $PC_i \in \mathbf{CS}_a$ and $PC_j \in \mathbf{CS}_b$ such that *PCi* is the CLCN PC of *PCJ*, while all other PCs are the same;
- if the number of PCs in $\mathbf{CS}_a$ is 1 less than that in $\mathbf{CS}_b$, and there is *only* 1 PPC in $\mathbf{CS}_b$ that is not contained in $\mathbf{CS}_a$.

Each contact state corresponds to a patch in C-space. For each patch, there is a range of values for each configuration variable. Using the definition of extremal configurations (in Section III.A) for a contact state and the relations among contact states, we have:

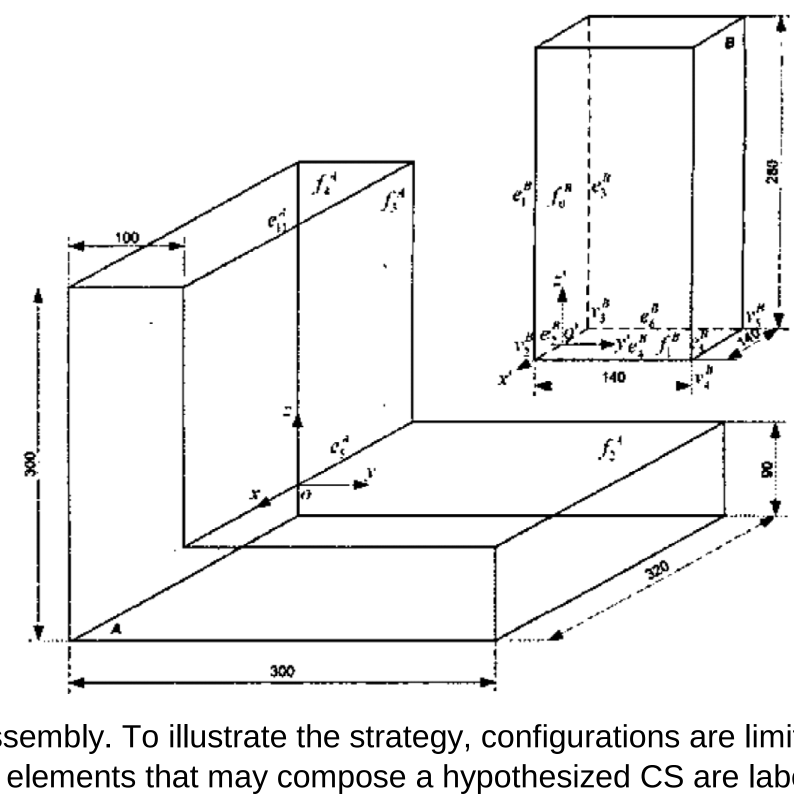

Fig. 1. Example Assembly. To illustrate the strategy, configurations are limited to small range. Only those surface elements that may compose a hypothesized CS are labeled.

Proposition 1

The configurations of a contact state cannot exceed the bounds defined by the minimum lower bound and maximum upper bounds of all its close less constrained neighbor contact states.

For a CS containing multiple PCs, the following proposition holds:

Proposition 2

The configurations of a contact state cannot exceed the bounds defined by the intersection of the range of each of the PCs contained in the CS.

B. Robustness Improvement

One parameter used in genetic algorithms is the number of individuals in the population. It determines the density of the distribution of individuals within the range of variables. For a given population size, a smaller range yields a greater population density in the solution space, and therefore is more likely to yield the true optimal solution.

There are two types of constraint on the range of configuration variables. One is determined by relative positioning inaccuracy. The other relates to restrictions associated with part geometry. The actual range of each variable for a contact state, in general, will be less than the initial range determined by robot inaccuracy.

Recall that the procedure for generating all contact states starts by considering less constrained contact states, after which the more constrained neighbor hypothesized contact states obtained from combinations of feasible PPCs are considered.

In the sequence of checking the feasibility of hypothesized contact states, the associated extremal configurations for a feasible contact state are determined prior to checking the feasibility of the next hypothesized contact state. Using *Proposition 1* and *Proposition 2*, these extremals are used to restrict the range of configuration variables. The feasibility of a new hypothesized contact state can be assessed within the range of configuration variables based on the extremal values of its CLCN contact states and PCs contained in the CS.

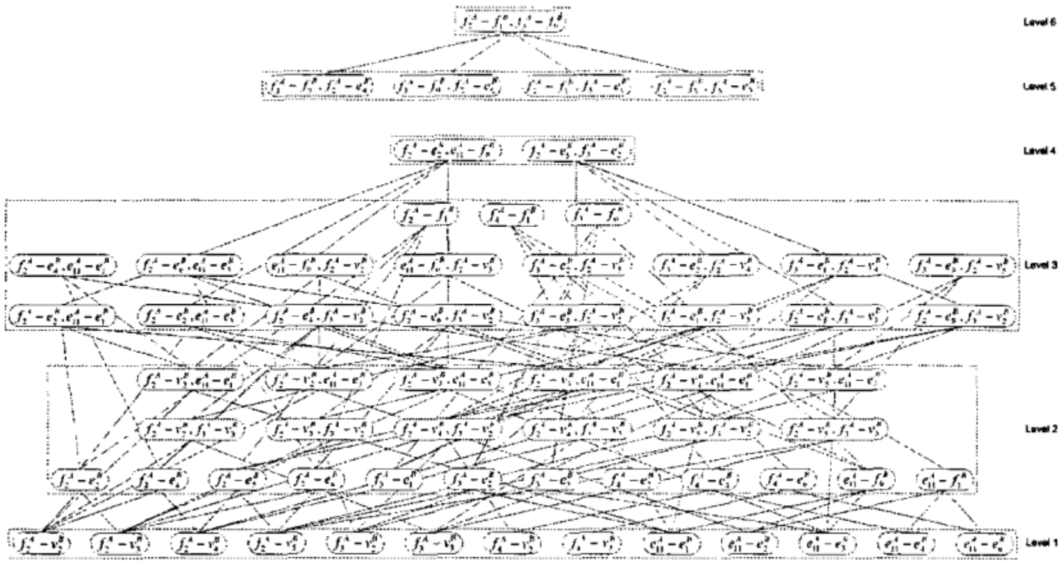

Fig. 2. ACSG for L-shape Assembly and Associated Close Neighbors.

Generally, this range is much smaller than the initial range imposed by robot inaccuracy (7). For example, if the range of each of the 6 configuration variables is half of its original range, the restricted search range will be 1/64 of the original range. By restricting the range of configuration variables, the density of the population is greatly increased and the robustness of the optimization is significantly improved at no addition cost in computation time.

The following describes the improved algorithm used to identify contact states and associated extremal configurations:

1. Generate set of PPCs from DXF files of the parts (A PPC is a PC having 5 degrees of freedom (face-vertex, vertex-face and edge-edge-cross));
2. Combine feasible PPCs to generate a hypothesized contact state;
3. Map the hypothesized contact state into the highest level topological description;
4. Obtain the bounds of the search area for the hypothesized contact state;
    1. Get all CLCN CSs for the hypothesized CS;
    2. Determine the lower and upper bounds for each variable for each of these CLCN CSs;
    3. Among all these lower (upper) bounds, find the minimum lower (maximum upper) bound;
    4. Determine the lower and upper bounds for each variable for each of the PCs contained in the hypothesized CS;
    5. Find the maximum lower (minimum upper) bound, among these bounds determined by PCs;
    6. Find the intersection of the ranges determined in (c) and (e).
5. Check the feasibility of the hypothesized contact state based on the refined bounds;
6. Obtain extremal configurations if the contact state is feasible;
7. Repeat Steps 2–6 until no additional PPC combinations can be generated.

Items 4(a)-4(c) identify the range determined by CLCN contact states based on *Proposition 1*. Items 4(d)-4(e) identify the range determined by PCs contained in the CS based on *Proposition 2*. If there are no CLCN for the hypothesized contact state, items 4(a)-4(c) are skipped and the range is determined only by 4(d)-4(e). If the hypothesized contact state consists of only 1 PC, 4(d)-4(e) are skipped and the range is determined only by 4(a)-4(c).

SECTION V.

Example

A simple example illustrates the improved means of generating all contact states and associated extremals. Figure 1 illustrates two parts and the coordinate frames used to indicate relative positioning. The misalignment bounds in this example are limited to a range consistent with the process of inserting the block into the L-shape corner from the top.

For inaccuracy imposed bounds given by:

$$\begin{aligned} \mathbf{t}_{min} &= [-5,-5,0,-\pi/20_3-\pi/20,-\pi/20]^T \\ \mathbf{t}_{max} &= [5,5,240,\pi/20,\pi/20,\pi/20]^T \end{aligned}$$

the generated ACSG shown in Fig. 2 is obtained. A total of 63 contact states are generated. Note that there are no CLCN or CMCN relationships between the contact states in level 4 and level 5. In this case, the level 5 contact states are derived from the combination of level 3 and level 1 (PPC) contact state. For this case, the search range is restricted by using the procedure described in items 4(d)-4(e) of Section IV.B.

To illustrate the ACSG generation strategy and the robustness improvement, consider the ACSG subset shown in Fig. 3. The lines between pairs of contact states indicate that the two contact states are close neighbor contact states. Figure 3 shows, for example, that $\{f_2^A-e_2^B\}$ is derived from the combination of $\{f_2^A-v_2^B\}$ and $\{f_2^A-v_3^B\}$ which are CLCN PCs. When the feasibility of $\{f_2^A-e_2^B\}$ is checked, the range of configurations is not the initial range between t*min* and t*max* determined by the robot uncertainty, but the range between

$$\begin{aligned} &\mathbf{t}'_{min} = \\ &[-5,9.88\times10^{-4},3.42\times10^{-3}, \\ &-8.80\times10^{-4},-0.157,-7.01\times10^{-2}]^T \\ &\mathbf{t}'_{max} = [5,5,10.95,1.97\times10^{-2},0.157,7.03\times10^{-2}]^T \end{aligned}$$

determined by its CLCN PCs, which is approximately 11200 of the initial range. Improvements at higher levels are even more dramatic, particularly, when the search dimension is decreased.

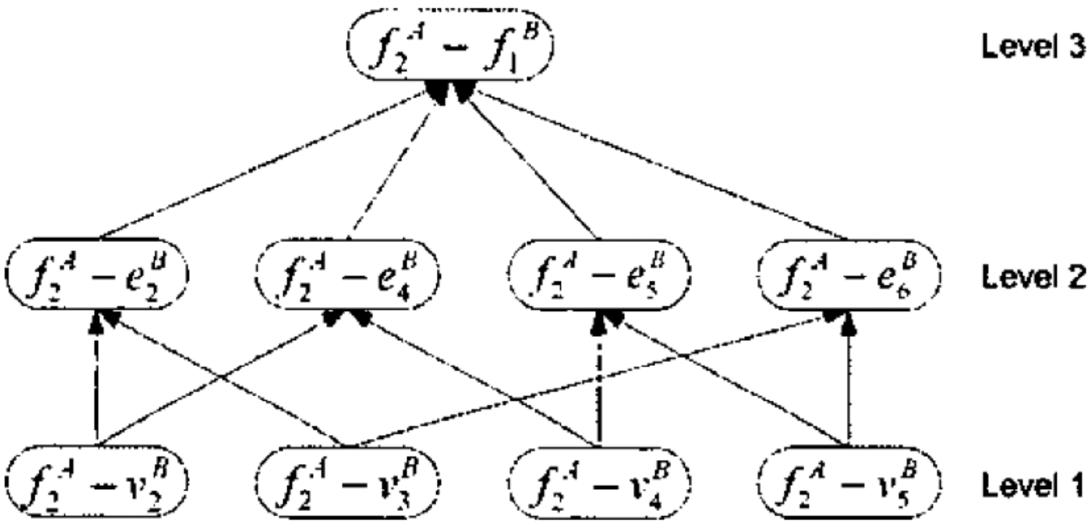

Fig. 3. ACSG subset. Arrows indicate the CMCN for each contact state generated. Levels indicate the level of kinematic constraint provided by contact and the contact state generation sequence.

Table I Assembly Contact States Evaluation and Ranges for Each Configuration Variable.

CS	Objective Value	Feasibility	Lower Bounds	Upper Bounds
$f_2^A - v_2^B$	7.37e-5	Feasible	[-5.0, 1.02e-3, 3.42e-3, -8.80e-4, 3.21e-3, -6.91e-2]	[5.0, 5.0, 10.93, 1.97e-2 , 0.157, 6.98e-2]
$f_2^A - v_3^B$	1.45e-4	Feasible	[-5.0, 9.88e-4, 5.61e-3, 8.86e-4 , -0.157, -7.00e-2]	[5.0, 5.0, 10.95, 1.95e-2 , 2.41e-3, 7.03e-2]
$f_2^A - v_4^B$	2.35e-4	Feasible	[-5.0, 9.92e-4, 1.03e-2, -0.151 , 9.66e-4, -6.98e-2]	[5.0, 5.0, 27.14, 1.04e-4 , 0.156, 6.85e-2]
$f_2^A - v_5^B$	4.12e-5	Feasible	[-5.0, 1.13e-3, 1.14e-2, -0.149, -0.155, -6.85e-2]	[5.0, 5.0, 27.07, 9.96e-5 , 1.22e-3, 6.78e-2]
$f_2^A - e_2^B$	5.83e-3	Feasible	[-5.0, 0.0, 1.06e-3, 2.38e-6 , 4.07e-5, -6.98e-2]	[5.0, 5.0, 3.02e-2, 1.79e-4 , 1.41e-4, 6.90e-2]
$f_2^A - e_4^B$	6.07e-3	Feasible	[-5.0, 8.86e-4, 3.35e-2, -7.38e-2 , 3.02e-4, -6.92e-2]	[5.0, 5.0, 10.81, 1.90e-2 , 0.155, 6.97e-2]
$f_2^A - e_5^B$	2.63e-3	Feasible	[-5.0, 1.34e-3, 3.35e-2, -1.31e-1, 1.05e-4, -6.87e-2]	[5.0, 5.0, 18.87, 2.47e-4 , 1.17e-4, 6.98e-2]
$f_2^A - e_6^B$	5.56e-4	Feasible	[-5.0, 1.45e-3, 1.71e-3, 1.96e-4, -0.157, -6.94e-2]	[5.0, 5.0, 10.88, 2.64e-4 , 3.05e-3, 6.92e-2]
$f_2^A - f_1^B$	7.93e-3	Feasible	[-5.0, 1.56e-3, 0.0, 1.02e-4 , 2.32e-3, -6.84e-2]	[5.0, 5.0, 4.36e-3, 2.76e-4 , 9.12e-3, 6.88e-2]
$f_4^A - v_4^B$	3.60e+3	Infeasible*	——	——
$f_3^A - e_4^B$	6.84e+1	Infeasible*	——	——
$f_4^A - f_6^B$	5.00e+1	Infeasible*	——	——
*Note: The assembly infeasible contact states listed in the table become valid contact states if there are no bounds on configuration.				

The values of the objective function for each hypothesized contact state within this subset and the extremal values for each configuration variable are summarized in Table I. Table I also indicates the values of objective functions for three infeasible contact states. Results show that the objective function values for feasible contact states are approximately zero. For infeasible contact states, the objective function has a relatively large positive number.

SECTION VI.

Discussion and Summary

In this paper, we have presented an approach for testing the assembly-feasibility of hypothesized contact states and for identifying extremal configurations of an assembly contact state using genetic algorithms. For the more constrained contact states, a search refinement strategy was presented to improve the optimization robustness using the extremal configurations and the neighbor relationships among the contact states.

ACKNOWLEDGMENT

This research is supported by the National Science Foundation under grant IIS 0010017.

References

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

[2] H. Hirukawa, Y. Papegay, T. Matsui, "A motion planning algorithm for convex polyhedra in contact under translation and rotation", *Proceedings of the International Conference on Robotics and Automation*, 1994.

[3] B. J. McCarragher, H. Asada, "The discrete event modeling and trajectory planning of robotic assembly tasks", *ASME Journal of Dynamic Systems Measurement and Control*, vol. 117, no. 3, 1995.

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

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

[6] S. Huang, J. Schimmels, "Sufficient conditions used in admittance selection for force-guided assembly of polygonal parts", *IEEE Transactions on Robotics and Automation*, vol. 19, pp. 737-742, August 2003.

[7] F. Pan, J. M. Schimmels, "Efficient contact state graph generation for assembly applications", *Proceeding of the IEEE International Conference on Robotics and Automation*, pp. 170-176, September 2003.

[8] J. Xiao, L. Zhang, "Computing rotation distance between contacting polyhedra", *Proceedings of IEEE International Conference on Robotics and Automation*, pp. 791-797, April 1996.

[9] X. Ji, J. Xiao, "Planning motions compliant to complex contact states", *The International Journal of Robotics Research*, vol. 20, no. 6, pp. 446-465, June 2001.

[10] C. Ong, 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.

[11] R. M. Z. Li, S. Sastry, A Mathematical Introduction to Robotic Manipulation, Boca Raton, Florida:CRC Press Inc., 1994.

[12] D. Goldberg, "Genetic Algorithms in Search" in Optimization and Machine Learning, New York:Addison-Wesley Pub Co., 1989.