Goal-contact Relaxation Graphs for Contact-based Fine Motion Planning

Jing Xiao
Computer Science Department
University of North Carolina - Charlotte
Charlotte, NC 28223, USA
xiao@uncc.edu

Abstract

In many robotic assembly tasks, fine motions are contact motions among parts which are inevitable because of uncertainties and are even desired to reduce uncertainties. Knowledge of the related contact states is extremely important in planning such motions. Representing contact states as contact formations *based on the notion of* principal contacts, *this paper introduces* goal-contact relaxation graphs *(GCR graphs) to represent and organize neighboring contact formations of typical "goal" contact formations. The GCR graphs carry concisely the information essential to contact motion planning and can be generated automatically. A task-independent algorithm for generating major subgraphs of GCR graphs is given in this paper.*

1 Introduction

Compliant motions and related subjects have become an important area of research since late 70's and early 80's, not only because many tasks require contact motions but also because contact motions have reduced degrees of freedom and thus reduced uncertainties. In addition, Hopcroft and Wilfong [7] proved that if two objects were in contact at two configurations and there was a way to move the objects from one of the contact configurations to another, then there was a way to move the objects between the two configurations through contact motions. Thus, contact motions are desirable in tight-clearance assembly tasks, and it is increasingly common to include contact motions deliberately in assembly motion planning. See, e.g., [9, 1, 14, 15, 2, 10, 13, 11].

While many planning systems incorporate certain knowledge of contact states, such knowledge is usually assumed *a priori* or generated manually as input to be used later in planning or in reasoning about sensory feedback (which often involves detection or recognition of contact states). Recently an algorithm was reported by Hirukawa *et al.* [6] to generate contact state graphs automatically. However, the study was limited to convex polyhedra and did not consider uncertainties. In reality, the presence of non-convex objects greatly increases the variety of contact states; manual enumeration of contact states can be too tedious even for the simplest assembly tasks, such as inserting a square peg into a square hole. Moreover, the presence of uncertainties even makes it a non-trivial matter how to represent a contact state properly (see Section 2 for discussion). There is clearly a need to study these issues of contact state representation and generation systematically, taking into account uncertainties.

In this paper, a concept of *goal-contact relaxation graphs* (GCR graphs) is introduced to organize the neighboring contact states of typical "goal" contact states of assembly tasks. A contact state is represented as a *contact formation* (CF) consisting of *principal contacts* (PC's) [16]. As detailed in Sections 2 and 3, such a representation of contact states is advantageous in the presence of uncertainties. A GCR graph of a goal contact formation CF_g captures the neighborhood of contact formations most likely to occur in the presence of uncertainties in the process of part-mating towards CF_g. It takes advantage of an important fact that those contact formations describing the goal states of an assembly task are often the most constrained contact formations in their certain neighborhoods. Such a neighborhood can then be obtained by "relaxing" the most constrained CF_g to be less constrained and can be represented by a GCR graph. Sections 4 discusses connectivity of CFs. Section 5 defines the GCR graphs, discusses their properties, and describes their generation. Section 6 concludes the paper.

2 Contact States: the Representation Issue

In the current literature, people describe contact states in a variety of ways based on their convenience, such as *topological representation* in terms of contacting topological surface elements, *force representation* in terms of contacting forces/torques, or *constraint representation* in terms of geometrical constraints on the locations of contacting objects. There is, however, little systematic and general investigation on the best way(s) of representing contact states.

The important questions we should ask are (1) what makes a contact different from another and (2) how easy it is to detect such difference and how much variations we can tolerate within one contact state, or in other words, what the granularity of representation should be. The two questions are essentially the same except that the second question is asked from a practical perspective, which is more relevant.

By referring to a "contact state", we usually mean a state of *higher* level than what the configurations of the contacting objects describe, since these objects may maintain the same *contact characteristics* while varing their configurations to some extent, and such common contact characteristics are often what we really care about.

However, most of these representations used are either *not* of levels high enough or too coarse. The most commonly used topological representation was the point-contact notion for polyhedra introduced by Lozano-Pérez [8] and Donald [5] in their attempt to construct C-obstacles. In this notion point contacts, for example, vertex-edge contacts in 2D polygons and vertex-face, edge-edge contacts in 3D polyhedra, were used as contact primitives, and a contact state was described as a set of such primitives. Many contact motion strategies assumed this representation. Another common topological representation was introduced by Desai *et al.* [4, 3] where they used the single contacts between a pair of topological surface elements of two polyhedra as primitives, called *elemental contacts* (ECs), and described a contact state (between the two polyhedra) in terms of the set of ECs formed, called a *contact formation*. From the viewpoint of contact recognition (i.e., sensing), both representations can result in states which are different by definition but indistinguishable in recognition due to uncertainties, as shown by the examples in Fig. 1.

Fig. 1a shows two objects, and Fig. 1b and c show two indistinguishable contact states between them due to uncertainties. By either the point-contact notion or the notion of ECs, these contact states are described differently. By point-contact notion, the contact state in Fig. 1b is described by v_3^A-f_1^B, f_1^A-v_1^B, e_2^A-e_1^B, e_3^A-e_4^B, but the contact state in Fig. 1c is described by v_1^A-f_1^B, v_2^A-f_1^B, v_3^A-f_1^B, v_4^A-f_1^B. By the notion of ECs, the contact state in Fig. 1b is described by v_3^A-f_1^B, f_1^A-v_1^B, e_2^A-e_1^B, e_3^A-e_4^B, e_2^A-f_1^B, e_3^A-f_1^B, f_1^A-e_1^B, f_1^A-e_4^B, f_1^A-f_1^B, but the contact state in Fig. 1c is described by v_1^A-f_1^B, v_2^A-f_1^B, v_3^A-f_1^B, v_4^A-f_1^B, e_1^A-f_1^B, e_2^A-f_1^B, e_3^A-f_1^B, e_4^A-f_1^B, f_1^A-f_1^B.

Using contact forces/torques to represent a contact state has the advantage of simplicity because contact states may be directly detected by the sensed forces/moments, but the force attribute *alone* as the state descriptor of a contact is often too coarse and does not facilitate symbolic reasoning in general. Similarly, although geometric constraints (equations or inequalities on the configurations of contacting objects) are extremely important properties of contacts, it is not convenient to use them *directly* as contact state descriptors [18].

3 Principal Contacts and Contact Formations

A good contact state descriptor should be a topological descriptor such that the state it describes is at the right level for use and for recognition (i.e., showing easily recognizable physical and geometrical properties). The concept of *principal contacts* (PC) first introduced in [16] holds such a promise. Denoting the *boundary elements* of a face as the edges and vertices bounding it, and the boundary elements of an edge as the vertices bounding the edge, a PC can be defined as the following:

Definition 1: A PC denotes the contact between a pair of surface elements (i.e., faces, edges, or vertices) which are not the boundary elements of other contacting surface elements (if there is more than one pair in contact).

Clearly, from the above definition, a PC is an EC (introduced previously), but not every EC is a PC. For polyhedral objects, there are ten types of PCs as shown in Fig. 2.

PCs are higher-level primitives and are less sensitive to location uncertainties of the contacting objects than ECs or the point-contact primitives (in the point-contact notion). For example, at the level of PCs, the indistinguishable contact situations in Fig. 1 are simply classified as belonging to the same contact state, described by the single PC f_1^A-f_1^B. Using PCs as contact primitives also simplifies geometric reasoning of contact constraints, as discussed in [18]. Thus, it is advantageous to describe a contact state in terms of the set of PCs formed, which we still call a *contact formation* (CF).

Moreover, we can practically discard degenerate PCs and *only consider the non-degenerate ones as contact primitives.* This is because among degenerate PCs, (1) the unstable ones formed by convex elements only (Fig. 2) hardly occur, and (2) the stable ones, each of which is formed by a convex and a concave element, can be treated as CFs of non-degenerate PCs, taking into account the roundness of edges/vertices and the small deformation due to contact in reality, as shown in Fig. 3. Note, however, for 2-D polygons, only the v-v type PC is degenerate. Considering CFs as sets of non-degenerate PCs also has an advantage from the perspective of contact forces/torques. Specifically, for a non-degenerate PC (Fig. 2), its change to another non-degenerate PC is often coincident with the discontinuity of the general contact force (i.e., force and torque), which means the change can be captured by force/torque sensing rather robustly in spite of sensing uncertainty.

Our experience in contact-based motion planning [14, 15, 17] shows that contact information at the level of PCs, together with the sensed location and force information, is often sufficient for tight-clearance assembly operations.

4 Neighboring Contact Formations

A general way of defining neighboring contact states was proposed by Desai [3] in terms of relative contact motions. Specifically, for objects A and B in contact, if there exists a contact motion which brings A and B in the contact state i to a new contact state j such that during the motion, A and B are always in contact and are not in any contact state other than states i and j, then i and j are called neighboring contact states. Desai's definition took advantage of the important result proven by Hopcroft and Wilfong [7] that if there was a way to move two objects from one contact configuration to another, then there was a way to do so with the objects remaining in contact throughout the motion. Desai's definition is very general: it applies to any kinds of contact state representations. However, how to use the definition, i.e., how to find *automatically* a proper contact motion in order to decide if two contact states are neighbors, is yet to be studied, and the issue is far from trivial. A particular awkwardness lies in the fact that motions are defined in the *continuous* configuration space, whereas the contact space is *discrete*.

On the other hand, it is highly desirable to map the general motion-based definition to a *topological* one in terms of how PCs (or the topological surface elements of the PCs) of neighboring contact formations are related. In fact, it is very easy to describe neighboring CFs (contact formations) between two convex polyhedra, where each CF consists of only a single PC. We give the definition below.

Definition 2: For two convex polyhedra A and B in contact, let $\mathrm{PC}_i = (a^A, b^B)$ and $\mathrm{PC}_j = (c^A, d^B)$ denote two PCs between them, represented by the pairs of contacting elements (i.e., faces, edges, or vertices). PC_i *contains* PC_j iff one of the following cases holds:

1. c^A is on the boundary of a^A and d^B is on the boundary of b^B,
2. c^A is on the boundary of a^A and d^B is b^B, or
3. c^A is a^A and d^B is on the boundary of b^B.

PC_i and PC_j are *neighboring PCs* if either one contains the other. If PC_i contains PC_j, then PC_j is the *less constrained* neighboring PC, under which the objects have more degrees of freedom, and PC_i is the *more constrained* neighboring PC.

With the above definition, the contact state space between two convex polyhedra can be built (or grown) from a particular PC by repeatedly finding the neighboring PCs. An example PC and some of its neighboring PCs are shown in Fig. 4.

If nonconvex polyhedra are involved, a CF may consist of more than one PC, and more complicated rules are required to describe neighboring CFs topologically. This is because geometric constraints on PCs which can co-exist need to be taken into account [18]. However, given a CF_i, it is relatively easier to characterize its *less-constrained* neighboring CFs to some extent:

General Rules

1. Any non-empty subset $\mathrm{CF}_j \subset \mathrm{CF}_i$ is a less-constrained[1] neighbor of CF_i.
2. For a $\mathrm{CF}_j \not\subset \mathrm{CF}_i$ with a cardinality less than or equal to that of CF_i, a necessary condition for it to be a less-constrained neighbor of CF_i is that every PC in CF_j is contained by a unique PC in CF_i (see **Definition 2**), and no two PCs in CF_j are contained by the same PC in CF_i.

Rule 2 describes a necessary condition because, although all such CF_js (which are valid[2]) are less-constrained CFs than CF_i, not all of them are neighbors of CF_i, and which of them are neighbors of CF_i depends on certain geometric features of objects. However, for CF_i and CF_j consisting of single PCs, Rule 2 describes a necessary *and* sufficient condition.

5 Goal-Contact Relaxation Graphs

As the set of goal configurations of an assembly usually either forms a single goal CF or can be partitioned by a limited number of goal CFs, it can greatly simplify representation and recognition to express the goal of an assembly in the contact formation space, i.e., in terms of the discrete goal CFs. In addition, knowledge of CFs in some neighborhood of the goal CFs can be very useful and even necessary in planning and executing fine motions towards to the goal CFs in the presence of uncertainty. Thus, there is a need to provide a planner a neighborhood of CFs of the goal CFs.

Observing that (one of) the goal CF(s), CF_g, is usually the most constrained CF within its certain neighborhood, we can describe that neighborhood by finding, in turn, the less-constrained neighboring CFs of CF_g, their less-constrained neighboring CFs, and so on, until no more less-constrained neighbors can be found. We can represent such a neighborhood of CF_g in a directed acyclic graph (dag) called the *goal-contact relaxation graph* (GCR graph) of CF_g. In a GCR graph, a node is a CF, and an arc connects two neighboring CFs,

[1]Here "less-constrained" is used weakly in that it also describes an "equally-constrained" (or *equivalent*) CF [18].

[2]Note that a set of PCs satisfying that every PC is contained by a PC of CF_i may not always form a valid CF.

pointing to the less-constrained one. There is a unique source node representing CF_g, and all sink nodes represent the least-constrained CFs, forming the boundary of the neighborhood. Fig. 5 shows the GCR graph of the goal CF of a 2D assembly.

A major subgraph $\mathcal{P}$ of a GCR graph can be easily generated following the **general rules** stated in Section 4 by the following algorithm based on breadth-first search:

Algorithm

1. Initialize $\mathcal{P}$ with one node CF_g and a FIFO queue *open* with one item CF_g.
2. If *open* is empty, goto 6.
3. Remove a node CF_x from open. If CF_x has a single PC, then
   - if it has no less-constrained neighboring PCs, goto 2, else, make every less-constrained neighboring PC a new CF;

   else, make every non-empty subset of CF_x a new CF.
4. For every new CF, CF_y, formed in Step 3, (1) if CF_y is not in $\mathcal{P}$, then add CF_y to *open* and to $\mathcal{P}$, (2) link CF_y in $\mathcal{P}$ to CF_x (as a less-constrained neighbor of CF_x).
5. Goto 2.
6. Return $\mathcal{P}$.

The subgraph $\mathcal{P}$ in the example of Fig. 5 consists of all the thicker arcs and the nodes linked by the thick arcs. Note that for certain transitions, such as the one from the top-middle CF to the top-right CF in Fig. 5, a degenerate PC (v-v) is encountered and discarded, which then make the two CFs neighbors.

To generate an entire GCR graph automatically requires the input of some specific geometric knowledge of the objects and the goal CF_g in addition to the **general rules** in Section 4. For example, it is not difficult to see that in Fig. 5, some of those neighboring CFs of CF_g not included in $\mathcal{P}$ may not be CF_g's neighbors if A is shorter than the vertical part of B or is wider than the horizontal part of B that it contacts. Nevertheless, if we classify CF_g's into different types, e.g., a CF_g of "2D or 3D corner assembly", and build rules specific to each type taking into account the effect of the relative dimensions of the contacting topological elements, an entire GCR graph can be automatically generated. Note that all CFs in a GCR graph are formed by only topological surface elements appeared in CF_g and/or their boundary elements (defined in Section 3), as clearly demonstrated in Fig. 5 by those thickened elements in CF_g. This property can greatly help in limiting the complexity of rules.

Even the subgraph $\mathcal{P}$ of a GCR graph can be valuable enough to fine motion planning but too complex (or large) to be generated manually (for 3D assembly in particular). So the **Algorithm** is useful. The most likely CFs that a fine motion towards CF_g can result due to uncertainties may be either in $\mathcal{P}$ or are neighbors of CFs in $\mathcal{P}$. If such a CF is (recognized to be) in $\mathcal{P}$, following the opposite directions of the arcs, a CF-space path can be found leading to CF_g; otherwise, by a simple transition to one of its neighbors in $\mathcal{P}$, a path to CF_g can be found. This, again, can be seen from Fig. 5.

6 Conclusions

This paper reviewed the advantages of representing contact states in terms of CFs with PCs as primitives and introduced the concept of GCR graphs in the CF space. The GCR graph of a goal contact formation CF_g of an assembly task can be very useful to fine motion planning, since it contains the CFs most likely to occur during a part-mating motion towards CF_g in the presence of uncertainties. When such a CF, CF_i, is identified, paths in the CF space connecting CF_i to CF_g can be searched along the opposite directions of the arcs in the GCR graph, and path optimization can be conducted based on certain criteria. An algorithm for automatic generation of a major subgraph of a GCR graph is presented. Further research on GCR graphs is necessary to fully explore its usage. Possible research directions include investigating rules with geometric knowledge for automatic generation of a whole GCR graph, the joining of different GCR graphs to cover greater space, and applications of GCR graphs.

References

[1] S. J. Buckley, "Planning Compliant Motion Strategies," *IEEE International Symposium on Intelligent Control*, pp. 338–343, August 1988.

[2] G. Dakin, and R. Popplestone, "Simplified Fine-Motion Planning in Generalized Contact Space," *IEEE Inter. Symp. on Intelligent Control*, pp. 281-287, 1992.

[3] R. Desai, *On Fine Motion in Mechanical Assembly in Presence of Uncertainty*, Ph.D. thesis, Department of Mechanical Engineering, the Univ. of Michigan, 1989.

[4] R. Desai, J. Xiao and R. Volz, "Contact Formations and Design Constraints: A New Basis for the Automatic Generation of Robot Programs", *NATO ARW: CAD Based Programming for Sensor Based Robots*, B. Ravani, Ed., Springer-Verlag, pp. 361-395, 1988.

[5] B. R. Donald, "On Motion Planning with Six Degree of Freedoms: Solving the Intersection Problems in Configuration Space," *IEEE Int. Conf. Robotics & Automation*, pp. 536-541, 1985.

[6] H. Hirukawa, Y. Papegay, and T. Matsui, "A Motion Planning Algorithm for Convex Polyhedra in Contact under Translation and Rotation," *IEEE Int. Conf. Robotics & Automation*, pp. 3020-3027, San Diego, May 1994.

[7] J. Hopcroft and G. Wilfong, "Motion of Objects in Contact," *Int. J. Robotics Res.*, 4(4):32-46, 1986.

[8] T. Lozano-Pérez, "Spatial Planning: A Configuration Space Approach," *IEEE Trans. Comput.*, C-32(2):108-120, 1983.

[9] T. Lozano-Pérez, M. T. Mason, and R. H. Taylor, "Automatic Synthesis of Fine-motion Strategies for Robot," *Int. J. Robotics Res.*, 3(1):3–24, Spring 1984.

[10] B. J. McCarragher, and H. Asada, "A Discrete Event Approach to the Control of Robotic Assembly Tasks," *IEEE Int. Conf. Robotics & Automation*, pp. 331-336, 1993.

[11] B. J. McCarragher, "Task Primitives for the Discrete Event Modeling and Control of 6-DOF Assembly Tasks," *IEEE Trans. Robotics and Automation*, 12(2):280-289, April 1996.

[12] M. A. Peshkin, "Programmed Compliance for Error Corrective Assembly," *IEEE Trans. Robotics and Automation*, 6(4):473-482, August 1990.

[13] R. H. Sturges, and S. Laowattana, "Fine Motion Planning through Constraint Network Analysis", *IEEE Int. Conf. Assembly and Task Planning*, pp. 160-170, Pittsburgh, August 1995.

[14] J. Xiao and R. Volz, "On Replanning for Assembly Tasks using Robots in the Presence of Uncertainties," *IEEE Int. Conf. Robotics & Automation*, pp. 638-645, May 1989.

[15] J. Xiao, "Replanning with compliant rotations in the presence of uncertainties," *IEEE Int. Sym. Intelligent Cont.*, pp. 102-107, Glasgow, U.K., August 1992.

[16] J. Xiao, "Automatic Determination of Topological Contacts in the Presence of Sensing Uncertainties," *IEEE Int. Conf. Robotics & Automation*, pp. 65-70, Atlanta, May 1993.

[17] J. Xiao and L. Zhang, "A Geometric Simulator for Testing the Replanning Approach toward Assembly Motions in the Presence of Uncertainties," *IEEE Int. Symp. Assembly & Task Planning*, pp. 171–177, Pittsburgh, August 1995.

[18] 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*, 13(3): 456–466, June 1997.

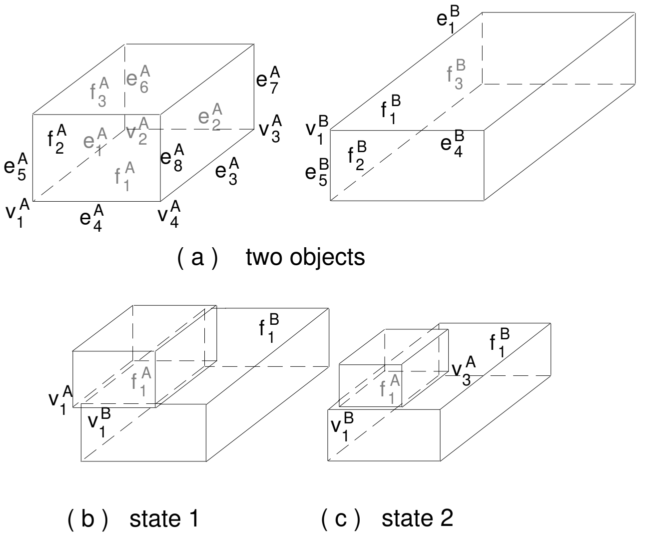

Figure 1: Indistinguishable states due to uncertainties

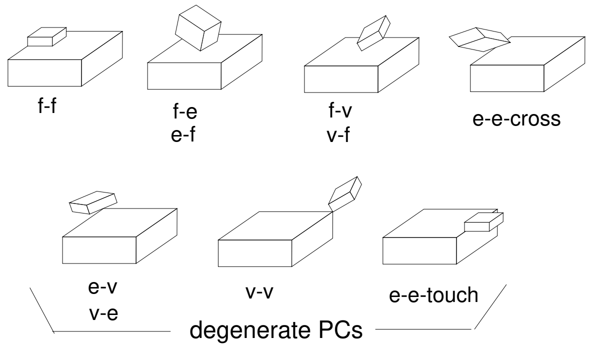

Figure 2: Principal contacts (PCs)

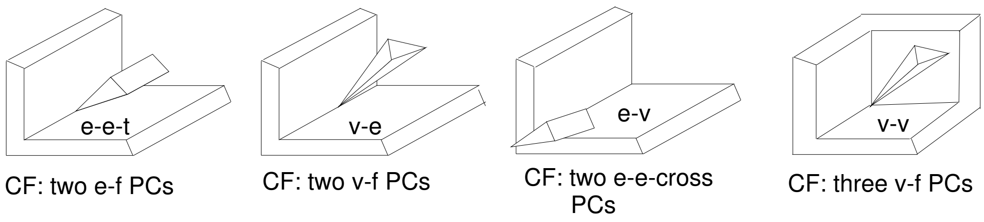

Figure 3: Degenerate but stable PCs are viewed as CFs rather than PCs

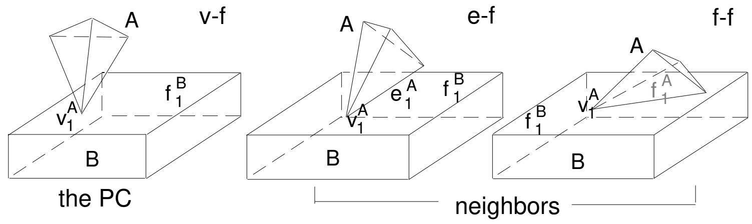

Figure 4: An example PC and some of its neighboring PCs

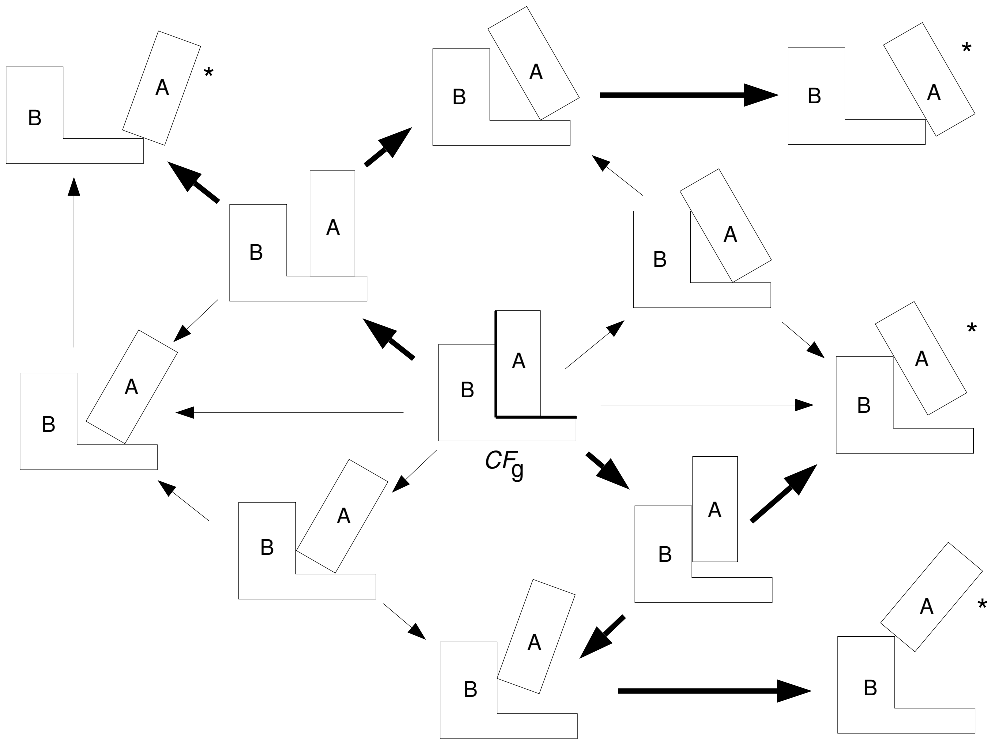

Figure 5: The GCR graph of CF_g of a 2D assembly. The CFs with $*$ are the least constrained CFs in the graph. The CFs and the thicker arcs linking them form the subgraph $\mathcal{P}$.