Regrasp Planning of Four-Fingered Hand for Parallel Grasp of a Polygonal Object

Thanathorn Phoka, Peam Pipattanasomporn, Nattee Niparnan, and Attawith Sudsang
Department of Computer Engineering
Chulalongkorn University
Bangkok 10330, Thailand
$\{$*phoka,peam.p,nattee,attawith*$\}$*@cp.eng.chula.ac.th*

***Abstract*— This paper proposes a necessary and sufficient condition for parallel grasps. We extend the use of this condition to the task of regrasp planning. In particular, we propose a graph structure called a switching graph which contains information about primitive grasping operations such as finger switching and finger sliding. The problem of regrasp planning is transformed to a graph search problem. Mainly, this work concentrates on a parallel grasp with force closure. Assuming frictional point contacts, the proposed method has been implemented and some preliminary results are presented.**

***Index Terms*— Regrasp, Parallel Grasp, Switching Graph**

I. Introduction

Grasping is an important task in robotics. Grasping an object is, however, not the ultimate goal of many problem. In the case that the current grasping configuration is not desirable, the contact points need to be repositioned. The ability to change contact points of a grasp formation while maintaining stability is clearly beneficial, especially for dexterous or in-hand manipulation. This work presents a method that plans the regrasp process of a polygonal object by a four-fingered hand. With four fingers, the hand can reposition its three-finger grasp by performing finger gaiting or switching. Our work concentrates on parallel grasps which are natural grasps for objects with parallel or almost parallel edges. We also find that feasible grasping configuration can be easily determined. The other benefactor is that we can perform finger sliding without difficulties.

The proposed method for regrasp planning is based on a structure called a switching graph. A necessary and sufficient condition for a parallel grasp is also proposed in this paper. The idea of a switching graph was originally presented in [1] for concurrent grasps and extended to cover 3-D grasps in [2]. A good review of in-hand and dexterous manipulation can be found in [3] and [4]. Finger gaiting was first suggested in [5]. A general framework for planning dexterous manipulation was presented in [6]. Omata and Nakata use branch-and-bound technique for planning a sequence of regrasps of a 2-D grasp [7]. Interested readers should referred to [3] for a recent review on robotic grasping.

The organization of this paper is as follows. The next section describes some background of grasping and proposes a new condition for a parallel grasp. Section III discusses a switching graph while section IV shows how to use a switching graph to actually plan a regrasp sequence. The experiment and its result is shown in section V. Finally, the paper is concluded in section VI.

II. Parallel Grasps

The scope of this work is limited to a 2-D grasping with friction by hard fingers. We assume Coulomb friction which allows a finger at each contact point to exert force only in a friction cone with half angle θ. An object being grasped is assumed to be polygonal. The property of grasping that we are interested in is the *Force Closure*. Force closure indicates that a grasp achieves equilibrium and is able to resist any external force/torque. Its properties are extensively studied in many literatures. Ponce and Faverjon [8] found that a stronger condition of equilibrium called nonmarginal equilibrium implies force closure. There are many kinds of equilibrium grasp but in this work we are only interested in a type of grasp, so called parallel grasps. The following proposition defines the parallel grasp.

Proposition 1: A necessary and sufficient condition for three points to form an equilibrium grasp with three parallel and non-zero contact forces is that there exists three parallel lines in the corresponding double-sided friction cone and for three vectors parallel to these lines and lying in the internal friction cone at the contact points, the vector parallel to the middle line are in the opposite direction from the other two.

Proof: Obviously, three parallel non-zero contact forces achieve a force equilibrium only when exactly one of them lies in the opposite direction of the other two. If the opposing force does not lie between the other two, the moment with respect to any points along the other vectors will not be zero. To achieve force closure, that force must be in the middle. In that case, it is obvious that a moment equilibrium can also be achieved. ■

Let $C_{\boldsymbol{n}_p}$ be a friction cone centered around a normal vector $\boldsymbol{n}_p$ of a contact point p. Proposition 1 indicates that two parallel forces must have the same direction while the other force must head to the opposite direction. For two contact points p and q, whose normals are $\boldsymbol{n}_p$ and $\boldsymbol{n}_q$, it is obvious that two parallel forces from these two contact points will exist only when the intersection of $C_{\boldsymbol{n}_p}$ and $C_{\boldsymbol{n}_q}$, both of them originated at the same point, is not empty. The existence of the third parallel force from contact point r which is in the opposite direction also indicates that the inverted cone $C_{-\boldsymbol{n}_r}$ intersects with the previous intersection. This condition is a necessary condition for a parallel grasp. Formally, there exists three parallel forces from three contact points p_1, p_2 and p_3 whose normals respectively are $\boldsymbol{n}_1, \boldsymbol{n}_2$ and $\boldsymbol{n}_3$ (Fig. 1(a)) when there exists $i, j, k \in \{1, 2, 3\}$ and $i \neq j \neq k$ such that the intersection of cones $C_{\boldsymbol{n}_i}, C_{\boldsymbol{n}_j}$ and $C_{-\boldsymbol{n}_k}$, all of them originated at the same point, is not empty (Fig. 1(b)). This condition is equivalent to the condition that the angle between $\boldsymbol{n}_i, \boldsymbol{n}_j$ and $-\boldsymbol{n}_k$ are pairwisely less than 2θ. If we limit θ to be less than $\pi/4$ (i.e., friction coefficient < 1), only one triple of (i, j, k) will satisfy the previous condition. In this work, we assume that θ is less than $\pi/4$. We call the contact point that has opposite direction force as a *center point*. We define a structure called a *common cone* that aids in existence checking of a parallel grasp as follows. A common cone exists only when three contact points p_1, p_2 and p_3 have three parallel forces, one of them lies in the opposite direction from the other two. From Fig.1(a), p_1 is the center point, a common cone $C^{\cap}_{\boldsymbol{n}_1,\boldsymbol{n}_2,\boldsymbol{n}_3}$ is the double-sided cone of the intersection of three cones $C_{-\boldsymbol{n}_1}, C_{\boldsymbol{n}_2}$ and $C_{\boldsymbol{n}_3}$ (Fig. 1(c)).

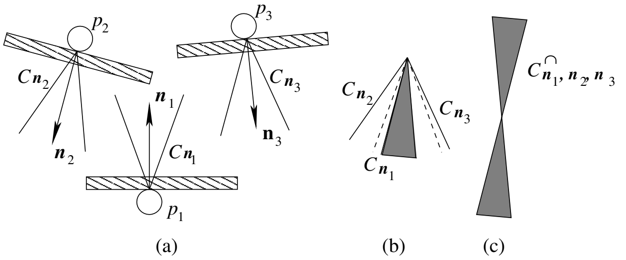

Fig. 1. Construction of a common cone : (a) A parallel grasp. (b) Three friction cones of (a) drawn at the same point. The dashed cone is inverted. (c) A common cone.

It is not straightforward to check the condition in Proposition 1 directly. We present a new condition for three contact points to form a parallel grasp as follows. If we draw a common cone on the center point, part of the plane that is not occupied by the cone will be divided into two regions. These regions are called the *outer regions*. The next proposition uses the notion of outer regions to define a necessary and sufficient condition for an existence of a parallel grasp when θ is less than $\pi/4$.

Proposition 2: A necessary and sufficient condition for three grasping points p_a, p_b and p_c, whose normals respectively are $\boldsymbol{n}_a, \boldsymbol{n}_b$ and $\boldsymbol{n}_c$, to form a parallel grasp is that two following conditions hold. (P_a) a common cone $C^{\cap}_{\boldsymbol{n}_a,\boldsymbol{n}_b,\boldsymbol{n}_c}$ is not empty. (P_b) Let us assume that the center point of these three points be p_a. The points p_b and p_c do not lie in the same outer region separated by the common cone $C^{\cap}_{\boldsymbol{n}_a,\boldsymbol{n}_b,\boldsymbol{n}_c}$ originated at p_a.

Proof: For the sufficient side, let us draw a segment connecting p_b and p_c. If both of them do not lie in the same side of the common cone, the segment $\overline{p_b p_c}$ will definitely intersect the common cone. Let p_x be any point in the intersection of $\overline{p_b p_c}$ and the common cone, we can draw a line from p_a to p_x. That line definitely lies in the friction cone of p_a (see Fig. 2). A line parallel to $\overline{p_a p_x}$ that passes p_b also lies in the friction cone of p_b, and so is the case of p_c. From a construction of a common cone, we can find three forces parallel along these lines that form a parallel grasp.

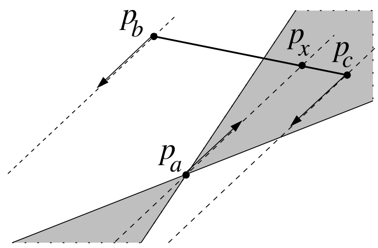

Fig. 2. When $\overline{p_b p_c}$ intersect with a common cone, we can find three parallel lines and vectors that satisfy 2

For the necessary side, if there exists a parallel grasp, a common cone will also exist. Now, if p_b and p_c lie in the same side, $\overline{p_b p_c}$ does not intersect the common cone. A line lying in the middle of p_b and p_c, which is necessary for a parallel grasp, must intersect with the segment $\overline{p_b p_c}$. However, since p_b and p_c lie completely in one outer region, every point in $\overline{p_b p_c}$ also lies in that outer region. It follows that if we pick some points on $\overline{p_b p_c}$ and use it to define a middle line, the other two lines passing through p_b and p_c that are parallel to the first line will also lie outside their respective common cone. Thus, at least one of them must lie outside its friction cone. This completes the proof as a contrapositive. ■

Since we are dealing with a polygonal object, we need a condition that can check the existence of a parallel grasp on polygonal edges. Proposition 2 can be extended to cover an existence of a parallel grasp on three polygonal edges. We define a union volume $U^a_{a,b,c}$ of polygonal edges a, b and c on edge a as the union of all common cones $C^{\cap}_{a,b,c}$ originated on every points of edge a. A union volume also divides the plane into two outer regions. We can uniquely identify the edge that contains a center point in the same way as the case of point. This edge will be called the center edge. Fig. 3 illustrates a union volume.

Proposition 3: A necessary and sufficient condition for the existence of a parallel grasp on three polygonal edges e_1, e_2 and e_3, whose normals are $\boldsymbol{n}_1, \boldsymbol{n}_2$ and $\boldsymbol{n}_3$, is that the two following conditions hold. (P_a) a common cone $C^{\cap}_{n_1,n_2,n_3}$ is not empty. (P_c) Let us assume that the edges

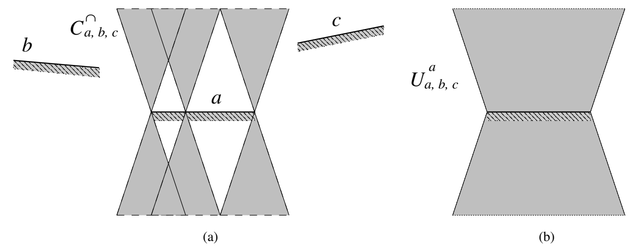

Fig. 3. (a) Three edges and a common cone drawn on some points on center edge. (b) A union volume.

that contain a center point is e_1. The edges e_2 and e_3 do not entirely lie in the same outer region separated by the union volume $U^{e_1}_{e_1,e_2,e_3}$.

Proof: Let p_2 be a point on e_2 and p_3 be a point on e_3. If two edges e_2 and e_3 do not entirely lie in the same outer region, then there exists p_2 and p_3 such that the line $\overline{p_2p_3}$ intersects with the union volume. According to the definition of the union volume, the point on the intersection of $\overline{p_2p_3}$ and the union volume must lie in a common cone of some point on e_1. Let us assume that the origin of that common cone is p_1. Three points p_1, p_2 and p_3 must form a parallel grasp according to Proposition 2. This complete the proof for the sufficient condition.

For the necessary side, since e_2 and e_3 entirely lie on the same outer region, every pair of point p_2 on e_2 and p_3 on e_3 also lies outside the union volume. From the definition of the union volume, we know that for every point p_1 on e_1, any pair of p_2 and p_3 will lie outside the common cone originated at p_1. From Proposition 2, we know that there can not be a parallel grasp. Thus, the proof is completed. ■

Now, we've already discussed the parallel grasp. The next section describes a regrasp process and proposes a structure that is used in the regrasp planning.

III. Switching Graph

Regrasp is a process of repositioning contact points of robot fingers. Two primitive forms of repositioning are *Finger Switching* and *Finger Sliding*. To determine an appropriate sequence of these two processes, we introduce a structure called a switching graph. A node in a switching graph represents a set of parallel grasps on three particular polygonal edges. An edge connecting two nodes indicates that there exists a grasp associated with one node that can be switched to a grasp associated with the other by finger switching. By using a switching graph, the regrasp problem can be formulated into a graph search problem. A path from the graph search determines a sequence of actions – switching and sliding to be executed in order to traverse from the initial to the final grasp. The following sections will describe the finger switching and sliding primitives and the switching graph in detail. The use of a switching graph is presented in section IV.

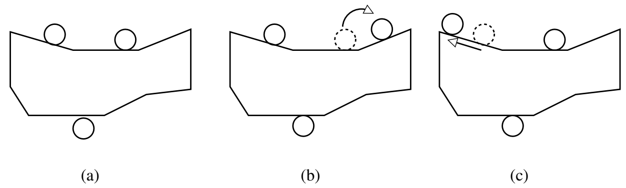

Fig. 4. (a) Initial grasping configuration (b) A result of finger Switching. (c) a result of a finger sliding

A. Finger Switching and Finger Sliding

Regrasp process which changes grasping configuration by placing an additional finger on desired contact point and then releasing one finger of the initial grasp is called finger switching. For example, let us assume that a starting grasp holds a polygonal object on points a, b and c and we want to switch to a grasp holding points b, c and d. A finger switching process starts by placing an additional finger on d and then releasing the finger at a. If both grasps satisfy the force-closure property, the entire process still holds the force-closure property. For the case of four-fingered hand grasping a polygonal object, finger switching requires that two grasping configurations must have two contact points in common and both of them achieve force closure. We call the two common points as non-switching contact points. Finger switching corresponds to an edge of switching graph. We will focus on this topic in section III-C.

Finger sliding is a process for repositioning fingers by sliding them along edges of a polygon while maintaining a force closure grasp during the sliding process. Using this process, we can change grasping configuration with in the same set of parallel grasps. This means the relation between finger sliding and a node of switching graph which will be explained in section III-B. However, finger sliding may be hard to implement mechanically since it is required that fingers must always touch the edge during sliding. Finger switching can imitate finger sliding by switching fingers from the initial to the final position of the sliding. Examples of finger switching and sliding are shown in Fig. 4.

B. Nodes in a Switching Graph

A node in a switching graph represents a set of parallel grasps on three particular edges. We denote $v_{a,b,c}$ as a node that applies on polygonal edges a, b and c. There will be a node $v_{a,b,c}$ in the graph if and only if there exists a set of parallels grasp on three edges a, b and c. In other words, each combination of three edges of a polygon will be associated with exactly one node. We use Proposition 3 to directly determine whether a node corresponding to a particular combination of edges exists in the graph or not. One grasping configuration can be changed to another by finger sliding when two of them are in the same node of the graph. We define a grasping set $G_{a,b,c}$ as a set of all grasping configurations represented as

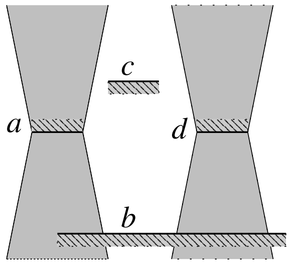

Fig. 5. (a) Unswitchable grasp (b) Switchable grasp

a triple of points (p_a, p_b, p_c) on polygonal edges a, b and c that form a parallel grasp. Each node in the switching graph corresponds to exactly one grasping set. Every grasp in each node can be repositioned to another grasp of the same node by finger sliding because of continuity in a set of parallel grasps for each triple of polygon's edges.

C. Edges in a Switching Graph

An edge linking two nodes, v_a and v_b, indicates that a finger switching can be performed between a grasping configuration in v_a and the other in v_b. Finger switching requires that two non-switching contact points must remain the same during the process. It follows that there will be an edge connecting two nodes when there are two grasping configurations, each of which belongs to each node, that use the same grasping points on the non-switching polygonal edges. Fig. 5 shows some examples of valid and invalid finger switchings. In Fig. 5(a), a grasp on edges a, b, c, where a is the center edge, can not switch to a grasp on edge b, c, d where d is the center edge. Edge c restricts graspable part of b to be inside the union volume of a for a grasp on (a, b, c). The same goes for a grasp on (b, c, d) which also restricts graspable part of b to be inside the union volume of d. These two areas are disjoint which mean that we cannot switch. In a contrary, Fig. 5(b) where a grasp on (a, b, c) (b as a center edge) can switch to a grasp on (b, c, d) (c as a center edge) because each pair of points on b and c is common for both grasps. Formally, there will be an edge connecting a node $v_{a,b,c}$ and a node $v_{b,c,d}$ when there exists a triple of points $(p_a, p_b, p_c) \in G_{a,b,c}$ and a triple $(p'_b, p'_c, p'_d) \in G_{b,c,d}$ such that $p_b = p'_b$ and $p'_c = p'_c$.

To check that we can switch between a grasping edge set $\{a, b, c\}$ and an edge set $\{b, c, d\}$, we compute two polytopes representing each grasping set. Let a be an edge with an end point $\boldsymbol{a}_0$ and a unit direction $\boldsymbol{t}_a$. The length of a is l_a. A point p_a on an edge a can be represented by $p_a = a_0 + u_a \boldsymbol{t}_a$ where $u_a \in [0, l_a]$. By using this representation, we can represent a set of all grasping configurations $G_{a,b,c}$ by a polytope in three dimensions, each dimension represents a value of u_a, u_b and u_c, respectively. Let P_1 be the polytope for edges $\{a, b, c\}$ and P_2 be the polytope for edges $\{b, c, d\}$. The space of P_1 and P_2

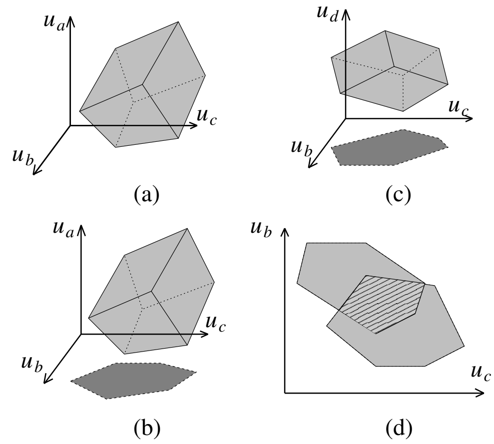

Fig. 6. (a) A polytope representing possible grasping points (in term of u_a, u_b and u_c (b), (c) two polytopes and their projections. (d) Intersection of the projected polygon representing a set of common points for a finger switching.

have two components (axes) in common, namely the axes of u_b and u_c. These components correspond to the non-switching edges, i.e., the common edges of both grasps. The projection of P_1 on the space of these two components represent the set of points on edges b and c that a parallel grasp on a, b and c is possible. Similarly, the projection of P_2 represents a set of points for a parallel grasp on b, c and d. If the intersection between these two projections is not empty, then there exists points on b and c that form a parallel grasp on both a, b, c and b, c, d. The reverse is also definitely true. Fig. 6 depicts the projection process.

The polytope P is defined by a set of linear constraints. For a polytope of a, b and c, a point is constrained to be on a polygonal edge. We define length constraints

$$0 \leq u_x \leq l_x \quad \text{for } x = a, b, c \tag{1}$$

Next, a set of constraints that bounds the contact point to satisfy Proposition 2 is presented. Let us assume that the center edge is a and the others are b and c. Intuitively, if one point p_b on b lies in an outer region (separated by a common cone of some point p_a on a), the second point p_c on c must be in a common cone of the third point or in the other outer region. However, the feasible area may not be convex so we construct it from a union of six convex polytopes. We define constraints for each of them as follows.

$$K_0 \equiv \begin{cases} \boldsymbol{n}_0 \cdot \overrightarrow{p_a p_b} \geq 0 \\ \boldsymbol{n}_1 \cdot \overrightarrow{p_a p_b} \geq 0 \\ \boldsymbol{n}_0 \cdot \overrightarrow{p_a p_c} \leq 0 \\ \boldsymbol{n}_1 \cdot \overrightarrow{p_a p_c} \leq 0 \end{cases} \tag{2}$$

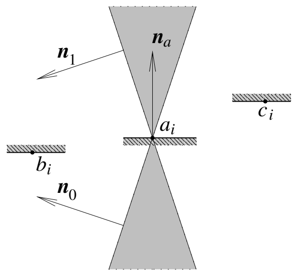

Fig. 7. A common cone on point a and vectors $\boldsymbol{n}_0$ and $\boldsymbol{n}_1$.

$$K_1 \equiv \begin{cases} \boldsymbol{n}_0 \cdot \overrightarrow{p_a p_b} \leq 0 \\ \boldsymbol{n}_1 \cdot \overrightarrow{p_a p_b} \leq 0 \\ \boldsymbol{n}_0 \cdot \overrightarrow{p_a p_c} \geq 0 \\ \boldsymbol{n}_1 \cdot \overrightarrow{p_a p_c} \geq 0 \end{cases} \tag{3}$$

$$K_2 \equiv \begin{cases} \boldsymbol{n}_0 \cdot \overrightarrow{p_a p_b} \geq 0 \\ \boldsymbol{n}_1 \cdot \overrightarrow{p_a p_b} \leq 0 \end{cases} \tag{4}$$

$$K_3 \equiv \begin{cases} \boldsymbol{n}_0 \cdot \overrightarrow{p_a p_c} \geq 0 \\ \boldsymbol{n}_1 \cdot \overrightarrow{p_a p_c} \leq 0 \end{cases} \tag{5}$$

$$K_4 \equiv \begin{cases} \boldsymbol{n}_0 \cdot \overrightarrow{p_a p_b} \leq 0 \\ \boldsymbol{n}_1 \cdot \overrightarrow{p_a p_b} \geq 0 \end{cases} \tag{6}$$

$$K_5 \equiv \begin{cases} \boldsymbol{n}_0 \cdot \overrightarrow{p_a p_c} \leq 0 \\ \boldsymbol{n}_1 \cdot \overrightarrow{p_a p_c} \geq 0 \end{cases} \tag{7}$$

Where $\boldsymbol{n}_0$ and $\boldsymbol{n}_1$ are the normal vector of left margin and right margin of the common cone respectively (see Fig. 7). The first two constraints, K_0 and K_1, are cases that u_b and u_c are on two distinct outer regions separated by a common cone at u_a while the others are for the others when u_b or u_c are the common cone. Each sub-polytope P'_i are defined as a convex hull constrained by Eqs. (1) and K_i.

When six sub-polytopes $\mathrm{P}'_0 \ldots \mathrm{P}'_5$ are constructed, we find its projection on a non-switching plane by examining its extreme points. For each sub-polytope, we project every extreme point of it on the non-switching plane and construct a convex hull from these points. The union of all projected convex hulls is a projection of the entire polytope.

At this point, a node and an edge in a switching grasp is described in detail. The next section describes the overall process of constructing a switching graph.

D. Computing a Switching Graph

To construct a switching graph, we start by building all nodes. Once all nodes are computed, every pair of nodes having two edges in common is checked for an edge by using a method described in section III-C. To save time, when a node is computed in the first step, we do a preprocessing of computing a projection of its polytope on all three pairs of planes (plane (u_a, u_b), plane (u_b, u_c) and plane (u_a, u_c)).

We compute all nodes by using a condition (P_a) in Proposition 3 for pruning. First, we sort every polygonal edge according to an angle between its normal and the x-axis. After that, we generate a triple of three edges that satisfies (P_a) and is checked against (P_c) to see whether it constitutes a node. The algorithm is shown in the following pseudo code. Let e_i represent i^{th} edge in the sorted list and m_i is an angle between the normal of e_i and the x-axis.

```
FOR i = 1 to n DO
    α = m_i
    j = (i + 1) mod n
    WHILE m_j < α + 2θ DO
        β = m_j
        FOR each k such that
            β + π − 2θ < m_k < α + π + 2θ DO
            IF e_i, e_j, e_k satisfy P_b THEN
                create a node for edges e_i, e_j, e_k
        j = (j + 1) mod n
```

First, we iteratively select an edge. The first edge limits that the angle of a normal vector between itself and a second edge must be less than 2θ. Once a second edge is selected, the angle of a third edge is also limited. Let α be the angle of a first edge and β be the angle of a second edge. The angle of a third edge must be in the open range of $(\beta + \pi - 2\theta, \alpha + \pi + 2\theta)$.

IV. Using Switching Graph

A switching graph provides a tool for planning a regrasp sequence. A path connecting the node containing the initial grasping position and the node containing the required grasping position indicates a sequence of edges that a finger switching should be performed. However, a path in a switching graph does not directly indicate which contact points on grasping edges are to be used in each step. For a pair of nodes having an edge connecting them, a switching graph tells us that we can switch between two grasps on these four edges but it does not tell which grasping points that we can do a finger switching. This section describe a method of transforming a path in a switching graph to an actual regrapsing sequence.

First, let us consider a finger switching. Finger switching takes place when we move from one node to another node in a graph. An edge in the graph tells us that a finger switching is viable. We have to find two grasps on each node that have two non-switching contact points in common. We randomly pick a point from the same intersection of the projections described in section III-C. That point indicates two actual points on non-switching edges. The next step is to find a point forming a grasp of the first node and a point forming a grasp of the second node. Let us consider a polytope defined in section III-C. Once a value of (u_b, u_c) in the intersection of projected P_1 and P_2 is chosen, we can construct a set of feasible contact points for the other two fingers by substituting u_b, u_c in (2) to (7) with the chosen values.

Next, let us consider a finger sliding. Finger sliding may be required in-between two finger switching, i.e., when we just traversed from node $v_{a,b,c}$ to node $v_{b,c,d}$ and about to move to the next node $v_{b,c,f}$. Let us assume that the first finger switching is just performed and we currently are in a grasp represented in $v_{b,c,d}$. In order to perform

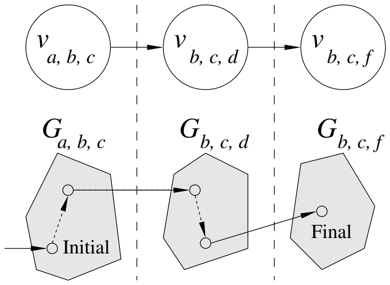

Fig. 8. A corresponding between nodes and edges in a switching graph and a finger switching and a finger aligning. A dashed line in the bottom figure represents a finger sliding while a solid line represents a finger switching

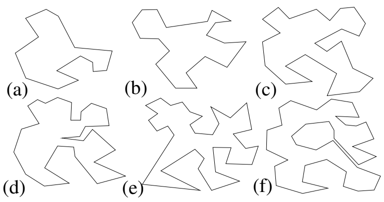

Fig. 9. Example polygons used in the experiment.

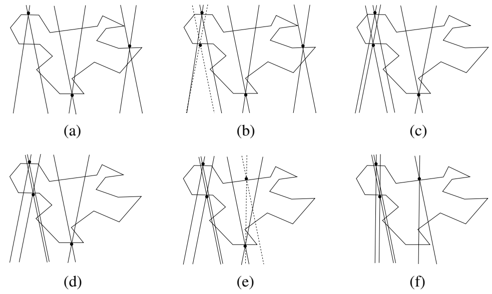

Fig. 10. (a)–(f) Actual regrasp sequence. A contact point with dashed cone is the new grasping point of a finger switching.

TABLE I

RESULT OF THE IMPLEMENTATION

Fig.	#Polygonal edge	Switching Graph		
		#node	#edge	build time(s)
9(a)	15	11	14	0.42
9(b)	20	31	74	0.72
9(c)	25	81	399	2.06
9(d)	30	62	202	1.88
9(e)	35	122	552	3.52
9(f)	40	249	1779	6.41

the next finger switching, i.e., to move to the node $v_{b,c,f}$, the grasping position must have two contact points in common with the final grasp. However, it might not be the current grasp. When an appropriate grasping configuration is computed as described earlier in this section, we have to change from the finishing grasp of the first switching to the a next switching. Since these two grasps lie on same polygon's edges, we can change the current grasp to an appropriate grasp for the next switching by a finger sliding. Fig. 8 shows the corresponding between a switching graph and the actual action performed on a regrasp.

V. EXPERIMENT AND RESULT

A program for planning a regrasp is implemented. The program uses the algorithm proposed in the section III-D for construction of a switching graph. Fig. 9 shows the testing polygons used in the experiment. The method in the section IV computes the actual sequence of regrasp actions. The time used for a computation of a switching graph and a regrasp action is measured. Table I shows details of each polygon and the time used by the implementation with half friction cone angle of 10 degrees. The column "build time" is the time used to construct a switching graph. The program was implemented in C++ and tested on Pentium IV 1.8GHz running Windows. Fig. 10 shows an actual sequence of regrasp process.

VI. DISCUSSION, CONCLUSION AND FUTURE WORK

We proposed a method for planning a regrasp of a polygon by extending the idea of the switching graph to cover a three-fingered parallel grasp. A necessary and sufficient condition for the existence of a parallel grasp is presented and used in the computation of a node in a switching graph. A method based on linear constraints is used to check a feasibility of a finger switching between two nodes. The work restricts that the friction coefficient is less than 1 which is quite general for most robot hands. By restricting the friction cone, we can eliminate what is usually not required for most robot hands.

We would like to extend our work to unite the switching graph for concurrent grasps, parallel grasps and 2-fingered grasps. The extension of the condition for a parallel grasp and a feasiblility of a parallel grasp finger switching to 3-D case are also interesting topics.

REFERENCES

[1] Attawith Sudsang and Thanathorn Phoka. Regrasp planning for a 4-fingered hand manipulating a polygon. In *IEEE Int. Conf. on Robotics and Automation*, pages 2671–2676, September 2003.

[2] Thanathorn Phoka and Attawith Sudsang. Regrasp planning for a 5-fingered hand manipulating a polyhedron. In *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, pages 3674–3679, October 2003.

[3] A. Bicchi and V. Kumar. Robotic grasping and contact: A review. In *IEEE Int. Conf. on Robotics and Automation*, 2000.

[4] A. Okamura, N. Smaby, and M. Cutkosky. An overview of dexterous manipulation. In *IEEE Int. Conf. on Robotics and Automation*, 2000.

[5] J.W. Hong, G. Lafferriere, B. Mishra, and X.L. Tang. Fine manipulation with multifinger hand. In *IEEE Int. Conf. on Robotics and Automation*, 1990.

[6] L. Han and J.C. Trinkle. Dextrous manipulation by rolling and finger gaiting. In *IEEE Int. Conf. on Robotics and Automation*, 1998.

[7] T. Omata and K. Nagata. Planning reorientation of an object with a multifingered hand. In *IEEE Int. Conf. on Robotics and Automation*, 1994.

[8] J. Ponce and B. Faverjon. On computing three-finger force-closure grasps of polygonal objects. *IEEE Transactions on Robotics and Automation*, 11(6):868–881, December 1994.