GRASP PLANNING UNDER UNCERTAINTY

A.M.Erkmen and H.E.Stephanou
School of Information Technology and Engineering
George Mason University
Fairfax, Virginia 22030

Abstract

This paper deals with the planning of dexterous grasps for multifingered robot hands operating in uncertain environments. We first describe a sensor-based approach to the planning of a *reach* path prior to grasping. We then develop an on-line, joint space finger path planning algorithm for the *enclose* phase of grasping. The algorithm minimizes the impact *momentum* of the hand. It uses a *Preshape Jacobian* matrix to map task-level hand preshape requirements into kinematic constraints. A master slave scheme avoids inter-finger collisions and reduces the dimensionality of the planning problem.

1 INTRODUCTION

The work described in this paper is motivated by applications that involve dexterous manipulation by autonomous or teleoperated robots in unstructured, uncertain environments. Examples may include equipment maintenance and repair operations in space, under the sea, in a nuclear power plant, or in a chemically contaminated area.

Robot manipulators have traditionally used a gripper (capable of opening and closing motions) attached to their wrist to achieve a rather modest level of mechanical dexterity. This has been adequate for simple manufacturing applications in which the environment may be conveniently structured. The need for a higher level of dexterity, more versatility and more adaptability in end-effectors has become increasingly apparent as the application of automation has grown into areas where the environment is unstructured, and tasks have become more complex. Multifingered hands hold a great deal of promise because of their ability to impart precise localized forces and velocities to objects, and because of their ability to provide stable grasps. Unfortunately, complications arise, as finger coordination, finger trajectory planning, and task planning are not well defined for multifingered hands.

The motion of a robot hand is subdivided into five phases [7]: (i) the reach phase during which the hand moves to the vicinity of some object, (ii) the preshape phase defines an approach volume between the fingers, (iii) the enclose phase until the object reaches the focus of the approach volume, (iv) the grip phase during which fingers apply forces to the object, (iv) the manipulation phase deals with the transfer of the available degrees of freedom to the object.

Our overall objective is to derive intelligent control algorithms for multifingered robot hands in unstructured environments. An outline of our overall approach to grasp planning is outlined in this section. Section two deals with more specific issues related to planning the reach path, while sections three and four focus on the derivation of a minimum momentum approach to finger path planning during the enclose motion of a preshaped hand.

Reach phase

Our main emphasis here is on the development of active sensing strategies. We have developed an *evidential classifier* [12] based on the concept of prototypes for the recognition of graspable objects from incomplete evidence. Prototypical objects and their possible interpretations are stored in a knowledge base during the off-line training stage. The output of the classifier is in the form of belief functions, and must therefore be disambiguated prior to grasping. A *disambiguation* scheme that minimizes the entropy [13] of the interpretation is discussed in section two of this paper.

Sensory data are first gathered off-line, and processed by the evidential classifier to determine a set of candidate reachable objects, or *targets* [15]. Targets are modeled as *attractors* in state space. Similarly, several sets of obstacles are identified, and also represented *repellers*. Each set of targets and repellers is assigned a weight corresponding to their entropy. These sets are used for the local (i.e. around the current position) planning of the reach path. During the execution of the planned motion, additional sensory data are gathered. This is done by using a Newton iteration method (discussed in section 3) that guides the hand closer to targets and obstacles with higher entropy. As additional and/or more refined data are acquired, the classification of targets and obstacles is updated on-line.

Preshape phase

The purpose of this phase is to preshape the hand into a configuration suitable for the anticipated action. Our work here is focused on a new theory of *prehensibility* [10] in which a topological model of prehension is used in conjunction with a knowledge based system to determine hand preshapes from a list of object properties and high-level task specifications. Objects are described geometrically (e.g. cylindrical shape), topologically (e.g. number of vertices, edges, faces), and functionally (e.g. used as a tool). Tasks are described in terms of geometrical, topological, and functional, and behavioral properties.

Enclose phase

The main focus of this paper is on the enclose phase. In sections 4 and 5, we describe a master-slave finger path planning algorithm during the enclose phase. Inputs to the algorithm include the hand preshape, and the desired cartesian position of the master fingertip. The algorithm generates a sequence of knot points (in joint space) that minimized the impact momentum of the master finger, while preserving the hand preshape constraints during the enclose motion. Our approach is based on a Newton iteration applied to the master finger. Using a dyadic expansion of the differential momentum of the master finger, we define a *Preshape Jacobian* that incorporates global (i.e. hand level) preshape constraints into the local (finger level) Newton scheme.
The method is illustrated by computing a Pinch Jacobian and a Hook Jacobian for 2D (planar) motion.

Grip phase

We have studied the performance of a *tentacle-based*, massively redundant manipulator [11] as an alternative to manipulation by an arm/hand combination. The tentacle manipulator is able to grasp by wrapping its links around an object in the same manner used by octopi. This method of grasping is advantageous because the tentacle becomes an all-in-one arm and gripping device capable of a variety of configurations and grasps, while utilizing the mechanics of serial manipulators. We have developed a quantitative method for the evaluation of grasp manipulability and stability accounts for multiple object contacts for each tentacle. Methods for applying both precision and power grasps to three dimensional objects for manipulation using a tentacle manipulator have been derived. These grasps are advantageous because each can be obtained from the other by merely curling or uncurling links from around an object, thereby reducing the number and complexity of grasp configurations.

2 SENSOR BASED REACHING

This section deals with object recognition and path planning during the reach phase of a dexterous grasp.

2.1 Minimum entropy disambiguation

An evidential classifier for object recognition has been described in [12]. We assume that low level sensory data processing has been completed, and that objects have already been detected in the segmented scene. The classifier uses shape primitives (e.g. rectangle, square, triangle) and matches them against an aggregate of prototypical graspable objects that are representative of all the classes (e.g. Pyramid, L-Shape, Handle, Cylinder) of interest. Because the sensory data as well as the aggregate of prototypes are generally incomplete, the classifier output is in the form of a belief function over the object frame of discernment, **FO**.

It is necessary to *disambiguate* the output of the classifier, i.e. to map the belief function in FO into a singleton (single object class), prior to grasping. This is done by using a minimum entropy criterion.

2.1.1 Algorithm

Let the output of the classifier be a belief function with core: $Q = \{q_1, \ldots, q_n\}$, and basic probability assignments: $B = \{b(q_1), \ldots, b(q_n)\}$.

The *class entropy* $h_c(\omega_i)$ of the ith object class is:

$$h_c(\omega_i) = \sum_{p_i \subseteq q_j} \left[-b(q_j)\log(b(q_j)) - \frac{p_i^*}{q_j^*}\log\left(\frac{p_i^*}{q_j^*}\right) \right]$$

The summation is over all focal elements q_j (in FO) containing ω_i. For a given object, the class entropies are computed for each class (singleton of FO). The object is then assigned to the class that yields the lowest entropy.

The belief function reflects two types of distribution among possible classes: (i) a topological distribution formed by the creation of focal elements corresponding to sets of classes, (ii) a probabilistic distribution of belief values assigned to the focal elements. Each class contributes to both types of distributions, and therefore to the generation of entropy. A class contributes to entropy in the topological distribution if it contributes to the confusion in choice. This occurs when a class is embedded in a focal element (i.e. set of classes). A class may also contribute to entropy because of the distribution of belief among focal elements.

2.1.2 Example

Let the belief function M denote the output of the evidential classifier [12]:

M = [{PYRD,LSHP,HNDL,(PYRD,LSHP)},{.09,.17,.4,.18}]

PYRD denotes the class of pyramid shaped objects, LSHP stands for L-shape objects, and HNDL represents the class of handle shaped objects.

The class entropy of PYRD is:

$$h_c(PYRD) = -b(q_1)\log(b(q_1)) - b(q_4)\log(b(q_4))$$

$$-\frac{PYRD^*}{PYRD^*}\log\left(\frac{PYRD^*}{PYRD^*}\right) - \frac{PYRD^*}{PYRD^* + LSHP^*}\log\left(\frac{PYRD^*}{PYRD^* + LSHP^*}\right)$$

$$= .23 + .16 = .39$$

The class entropy of LSHP is:

$$h_c(LSHP) = -b(q_2)\log(b(q_2)) - b(q_4)\log(b(q_4))$$

$$-\frac{LSHP^*}{LSHP^*}\log\left(\frac{LSHP^*}{LSHP^*}\right) - \frac{LSHP^*}{PYRD^* + LSHP^*}\log\left(\frac{LSHP^*}{PYRD^* + LSHP^*}\right)$$

$$= .27 + .13 = .40$$

The class entropy of HNDL is:

$$h_c(HNDL) = -b(q_3)\log(b(q_3)) - \frac{HNDL^*}{HNDL^*}\log\left(\frac{HNDL^*}{HNDL^*}\right) = .16 + 0 = .16$$

The object is classified as a handle since the HNDL class has the lowest entropy.

2.2 Reach path planning

In this section, we assume that a set of targets (Eg. HNDL) and several sets of obstacles (e.g. PYRD and LSHP) have been recognized. Our goal is to design an on-line path planning algorithm for the reach phase. This algorithm can adapt to updates in the classification of targets and obstacles as additional sensory data are gathered.

2.2.1 Target and obstacle representation

Our approach to path planning is based on a local rather than global strategy. To accomplish this goal, we generate two sets of vector polynomial functions: the *target function* $H^A(X)$ and the *obstacle functions* $H^{R_j}(X)$ such that

$$H^A(X_i^a) = 0 \ , \ H^{R_j}\left(X_i^{o_j}\right) = 0$$

where X is the state vector, X_i^a is the location of the ith attractor (target), and $X_i^{o_j}$ is the location of the ith repeller (obstacle) in the jth obstacle set.

2.2.2 Algorithm

Our approach to path planning is based on the Newton iteration:

$$X_{k+1} = X_k - \frac{\delta_k^a}{\gamma^a}[\nabla H^a(X_k)]^{-1} H^a(X_k)$$

$$+ \sum_j \frac{\delta_k^{\rho j}}{\gamma^{\rho j}}\left[\nabla H^{\rho j}(X_k)\right]^{-1} H^{\rho j}(X_k)$$

where γ^a is the class entropy for the set of attractors, $\gamma^{\rho j}$ is the class entropy for the jth set of repellers, and δ_k^a , $\delta_k^{\rho j}$ are weighting coefficients discussed in [15].

2.2.3 Example

Assume 4 HNDL attractors at

$X_1^a = [1\ 2]\ X_2^a = [3\ 1]\ X_3^a = [4\ 2]$

3 PYRD repellers at

$X_1^{\rho 1} = [1.5\ 3]\ X_2^{\rho 1} = [2.5\ 2]\ X_3^{\rho 1} = [3.5\ 3.5]$

and 4 LSHP repellers at

$X_1^{\rho 2} = [1\ 1]\ X_2^{\rho 2} = [2\ 4]\ X_3^{\rho 2} = [4\ 3]\ X_4^{\rho 2} = [5\ 2]$

Fig. 1a shows the *field* created by the targets and obstacles. Fig. 1b shows the trajectories for two sets of target and obstacle class entropies. Repellers are represented by filled-in squares, and attractors by filled-in circles. Initial states are indicated by empty circles. Trajectories labeled T_1, T_2, etc. are generated with $\gamma^a = 1$, $\gamma^{\rho j} = 1.25$, while trajectories labeled S_1, S_2, etc. are generated by assigning lower entropy for the attractors, and higher entropy for the repellers, namely: $\gamma^a = .5$, $\gamma^{\rho j} = 2$. In this case, the trajectories pass closer to the obstacles.

3 FINGER PATH PLANNING

In this section, we apply a Newton iteration similar to the one described in section two to path planning for a single finger during the enclose phase. The results are then extended in section 4 to the grasping motion of a two-fingered hand, by using a master-slave scheme. Our algorithm is based on a Newton iteration scheme that generates a sequence of knot points (in joint space) through which the master finger must pass. This scheme minimizes the *impact momentum* of the finger, evaluated at the desired fingertip contact location.

3.1 Finger momentum

When a wrench vector is applied to the ith finger, it causes changes in its momentum vector G_i. Let r_i denote the fingertip position vector, v_i denote the fingertip velocity, and Δr_i denote a finite fingertip displacement, all in cartesian coordinates relative to a base (palmar) frame. Similarly, let θ_i and ω_i denote the vectors of joint angles and velocities for the master finger, and $\Delta\theta_i$ denote a finite finger displacement in joint space.

The *momentum* of the ith finger is given by :

$G_i = m_i v_i \times r_i + m_i v_i$

where $m_i v_i \times r_i$ is the angular momentum of the finger, $m_i v_i$ is its linear momentum, and m_i is its mass, assumed to be concentrated at the fingertip.

3.2 Finger path planning

The differential mapping of Δr_i into momentum changes ΔG_i is given by:

$\Delta G_i = J_{r_i} \Delta r_i = J_{r_i} J_{f_i} \Delta \theta_i = J_{\theta_i} \Delta \theta_i$

where J_{f_i} is the finger Jacobian, i.e. $\Delta r_i = J_{f_i} \Delta \theta_i$

Our approach to finger path planning is based on the minimization of its impact momentum. Let θ_i^* denote the desired fingertip position. To find the roots of the momentum function $G_i(\theta_i - \theta_i^*)$, we generate a sequence of knots points by the Newton iteration :

$\theta_i^{k+1} = \theta_i^k - J_{\theta_i}^{-1} G_i(\theta_i^k - \theta_i^*)$

3.3 Dyadic expansion

Since the finger momentum is a vector quantity, the iterative procedure used for its minimization requires the expansion of the differential momentum in its *dyadic* form (reference), i.e.:

$\Delta G_i = (\Delta r_i \cdot \nabla) G_i$

$$= \frac{1}{2}[\nabla \times (G_i \times \Delta r_i) + \nabla (G_i \cdot \Delta r_i) + \Delta r_i (\nabla \cdot G_i) - G_i \times (\nabla \times \Delta r_i) - \Delta r_i \times (\nabla \times G_i)]$$

For the special case of 2D grasping in a plane, the angular momentum does not lie in the plane of motion. The linear and angular components of the momentum are therefore not additive. Instead, we use the planar discrepancy between angular and linear momentum (for unit mass), i.e.

$G_i = (v_i \times r_i) \times v_i$

The expansion of the dyadic G_i yields :

$$(\Delta r_i \cdot \nabla) G_i = (\Delta r_i \cdot \nabla)(v_i \times r_i) \times v_i$$

$$= \frac{1}{2}\{ [(v_i \cdot \Delta r_i) r_i - (r_i \cdot v_i)](\nabla \cdot v_i)$$

$$+ [(v_i \times r_i) \cdot \Delta r_i](\nabla \times v_i)$$

$$+ (r_i \cdot v_i)[v_i \times (\nabla \times \Delta r_i) + \Delta r_i \times (\nabla \times v_i)]$$

$$- (v_i \cdot v_i)[\Delta r_i \times (\nabla \times r_i) + r_i \times (\nabla \times \Delta r_i)]$$

$$+ \nabla[(v_i \cdot v_i)(r_i \cdot \Delta r_i) - (r_i \cdot v_i)(v_i \cdot \Delta r_i)]$$

$$- (v_i \cdot \Delta r_i)(V_i r_i - R_i v_i + 2 v_i) + 2(v_i \cdot v_i) \Delta r_i \}$$

where

V_i and R_i are the Jacobian matrices of the functions v_i and r_i respectively.

The first line in the dyadic expansion consists of divergence terms. The next three lines consist of curl terms, while the last two lines only contain terms that are not differential.

We postulate that hand preshapes can be analyzed in terms of two motion characteristics:

(i) the *hand flux*, and
(ii) the *hand curl*

Figure 2 shows a schematic representation of a three fingered hand. The hand encloses a *volume* bounded by the fingers. The hand flux is the sum of the divergence of its N fingers moving over a shrinking preshape volume:

$$(\nabla \cdot \mathbf{v})^H = \sum_{i=1}^{N} (\nabla \cdot \mathbf{v}_i)$$

As the fingers close during grasping, the fingertip moves along a path enclosing a surface that shrinks. The curl vectors define the directional curling of the finger with respect to its own base, and are given by:

$$(\nabla \times \mathbf{r})^H = \sum_{i=1}^{N} \nabla \times \mathbf{r}_i$$

$$(\nabla \times \mathbf{v})^H = \sum_{i=1}^{N} \nabla \times \mathbf{v}_i$$

4 MINIMUM MOMENTUM GRASPING

Different types of grasping motion are normally associated with the different preshapes of the hand. In this section, we derive a *Preshape Jacobian* matrix for the preshapes of a 2-fingered hand moving in the plane. Our goal is to map high-level task specifications into joint angles and velocities.

4.1 The Preshape Jacobian

In this section, we modify the Newton scheme described in section 3.2 to include the global (hand level) preshape constraints embedded in the Preshape Jacobian. The modified iteration,

$\theta_m^{k+1} = \theta_m^k - J_\pi^{-1} G_m(\theta_m^k - \theta_m^*)$

also minimizes the momentum of the master finger (i=m), but uses the matrix J_π which we call the *Preshape Jacobian* instead of the matrix J_{θ_i}. The Preshape Jacobian is defined by:

$$J_\pi \Delta\theta_m = \Delta G_\pi$$

ΔG_π is the *preshape momentum differential*. It incorporates global preshape constraint information into the path of the master finger.

ΔG_π is obtained from ΔG_i by replacing the flux and curl terms for a single finger by the flux and curl expressions for the whole hand, i.e. replace

$$(\nabla \cdot \mathbf{v}_i) \quad \text{by} \quad (\nabla \cdot \mathbf{v})^H = \sum_{i=1}^{N} \sigma_i^{\bullet v} (\nabla \cdot \mathbf{v}_i)$$

$$(\nabla \times \mathbf{v}_i) \quad \text{by} \quad (\nabla \times \mathbf{v})^H = \sum_{i=1}^{N} \sigma^{\kappa v} (\nabla \times \mathbf{v}_i)$$

$$(\nabla \times \mathbf{r}_i) \quad \text{by} \quad (\nabla \times \mathbf{r})^H = \sum_{i=1}^{N} \sigma_i^{\kappa r} (\nabla \times \mathbf{r}_i)$$

where the coefficients:

$-1 \le \sigma^{\phi v}, \sigma^{xv}, \sigma^{xr} \le 1$

depend on the specific hand preshape constraint. Two examples are given below.

4.2 The Pinch Jacobian

In the pinch grasp (fig. 3a), both fingers contribute to grasping. We assume the two fingers are preshaped symmetrically, and that they remain symmetric during the reach motion. The following set of constraints is used to model the pinch preshape:

$$(\nabla \times v)^H = 0 \qquad (\nabla \times r)^H = 0 \qquad (\nabla \cdot v)^H = \nabla \cdot v_1 + \nabla \cdot v_2$$

and leads to the *Pinch Jacobian*:

$$\Delta G_R = J(\theta, \omega)_{R_{pinch}} \Delta \theta_m$$

4.3 The Hook Jacobian

For the hook grasp (fig. 3b), fingertip 2 is coupled with joint j of finger 1. The constraints are:

$$r_{2_x} = r_{j_x} \qquad r_{2_y} = -r_{j_y} \qquad v_{2_x} = v_{j_x} \qquad v_{2_y} = -v_{j_y}$$

and lead to:

$$(\nabla \times r)^H = \nabla \times r_2 - \nabla \times r_j \qquad (\nabla \times v)^H = \nabla \times v_2 - \nabla \times v_j \qquad (\nabla \cdot v)^H = \nabla \cdot v_2 + \nabla \cdot v_j$$

The *Hook Jacobian* is determined from:

$$\Delta G_R = J(\theta, \omega)_{R_{hook}} \Delta \theta_m$$

5 DISCUSSION

The minimum momentum grasp planning described in this paper was motivated by applications such as NASA's EVA Retriever, which is required to grasp loose objects tumbling freely in space. In our algorithm, one finger is designated as the master, and its path is planned so as to minimize the impact momentum. A Preshape Jacobian was derived to map task-dependent preshape constraints into kinematic constraints, and thus provide the necessary coupling with the slave fingers. Planning the paths of the slave fingers follows directly from these constraints. We are currently conducting computer simulations for various 2D grasps. The concept of Julia sets [2] is used to graphically study the convergence of the grasping process in various regions of the state space. We are also in the process of deriving expressions of four (fingertip, lateral pinch, cylindrical, and hook) 3D Preshape Jacobians for the Stanford/JPL hand.

6 REFERENCES

1. M.R.Cutkosky and P.K.Wright, 1986, "Modeling manufacturing grips and correlations with the design of robotic hands," *Proc. IEEE Conf. Robotics and Automation*, pp. 1533-1539
2. R.L.Devaney, 1987, *Chaotic Dynamical Systems*, Addison Wesley
3. A.M.Erkmen and H.E.Stephanou, 1987, "Evidential control of dextrous grasps," *Proc. IEEE Conf. Systems Man and Cybernetics*, pp.583-587
4. P.Hsu, Z.Li, and S.Sastry, "1988, On grasping and coordinated manipulation by a multifingered robot hand," *Proc. IEEE Conf. Robotics and Automation*, pp. 384-389

5. T.Iberall, 1987, "The nature of human prehension: three dextrous hands in one," *Proc. IEEE Conf. Robotics and Automation*, pp. 396-401
6. Z.Li and S.Sastry, 1987, "Task oriented optimal grasping by multifingered Robot Hands," *Proc. IEEE Conf. Robotics and Automation*, pp.389-394
7. D.M.Lyons, 1985, "A simple set of grasps for a dextrous hand," *Proc. IEEE Conf. Robotics and Automation*, pp.588-593
8. D.M.Lyons, 1986, "Tagged potential fields: an approach to specification of complex manipulator configurations," *Proc. IEEE Conf. Robotics and Automation*, pp.1749-1754
9. J.R.Napier, 1956, "The prehensile movement of the human hand," *J. Bone Joint Surgery*, Vol.38B, pp.902-913
10. T.N.Nguyen and H.E.Stephanou, 1989, "A topological model of multifingered prehension," *IEEE Int. Conf. Robotics and Automation*
11. J.S.Pettinato and H.E.Stephanou, 1989, "Manipulability and stability of a tentacle based robot manipulator," *IEEE Int. Conf. Robotics and Automation*
12. H.E.Stephanou and A.M.Erkmen, 1988, "Evidential classification of dexterous grasps for the integration of perception and action," *Journal of Robotic Systems*, Vol.5, No.4, pp.309-336
13. H.E.Stephanou and S.Y.Lu, 1988, "Measuring consensus effectiveness by a generalized entropy criterion," *IEEE Trans. Pattern Analysis and Machine Intelligence*, Vol. 10, No.4, pp.544-554
14. R.Tomovic, G.A.Bekey and W.J.Karplus, 1987, "A strategy for grasp synthesis with multifingered robot hands," *Proc. IEEE Conf. Robotics and Automation*, pp.83-89
15. F.Yegenoglu, A.M.Erkmen, and H.E.Stephanou, 1988, "On-line path planning under uncertainty," *Proc. IEEE Conf. Decision and Control*

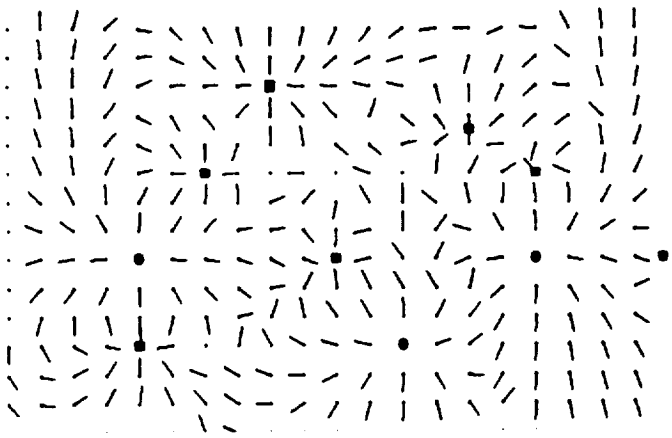

Fig.1a Field

Fig.1b Trajectories

Fig.1 Reach Path Planning

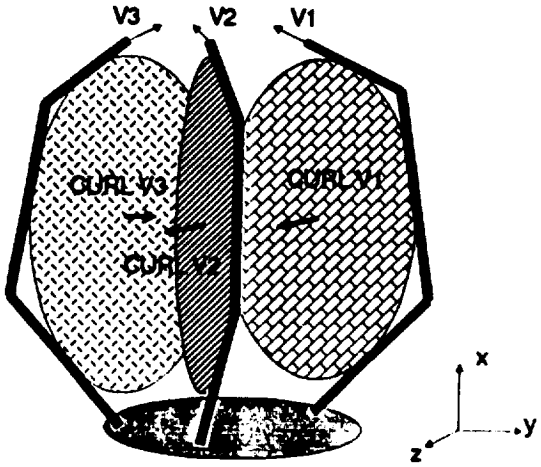

Fig. 2a Link Slices

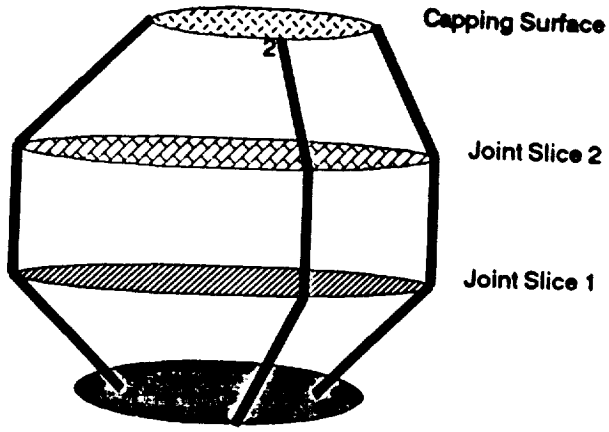

Fig. 2b Joint Slices

Fig. 2 Volumetric Model

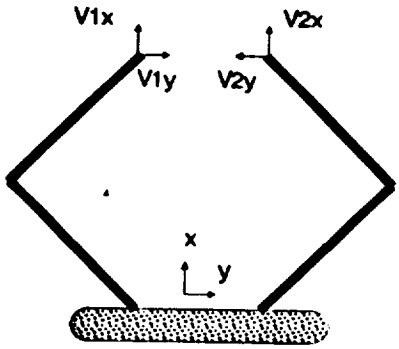

Fig.3a 2D Pinch Grasp

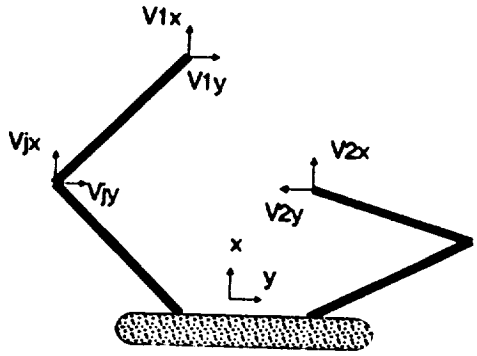

Fig.3b 2D Hook Grasp

Fig. 3 2D Preshapes