Randomized Manipulation Planning for A Multi-Fingered Hand by Switching Contact Modes

Masahito Yashima Yoshikazu Shiina Hideya Yamaguchi Dept. of Mechanical Systems Engineering, National Defense Academy Yokosuka 239-8686, Japan

Abstract—This paper presents a randomized manipulation planner for a multi-fingered hand by switching contact modes. Manipulation planning for such a system should consider changing kinematics and dynamics of the system according to the contact modes. We derive the conditions that would be satisfied in manipulation planning, based on the kinematical properties of a manipulation system. The conditions allow the planner to assign more contact modes than those corresponding to the relative motion of contact points restricted by constraint forces. Consequently, this enables dextrous manipulation. In addition, the conditions give the restrictions of feasible contact modes and the number of contact points. Inspired by randomized motion planning techniques, we propose a new algorithm for manipulation planning in order to explore object configuration space, SE(3), rapidly and uniformly. The basis for this approach is for the construction of exploring random trees. Simulation examples for 3-D manipulation by switching contact modes are presented to verify the planner's effectiveness.

I. INTRODUCTION

Manipulating an object by a hand, we dextrously use different contact modes such as rolling and sliding contacts unconsciously. Utilizing different contact modes according to a manipulation task can raise manipulation skill with the hand. The primary motivation of our work is to manipulate the object by a multi-fingered robot hand utilizing different contact modes.

Work on manipulation by a multi-fingered robot hand has attracted attention. For example, there is work on manipulation using rolling and/or sliding contacts [2], [3], [7], [9]. Most of the work has been devoted to the instantaneous kinematic and dynamic analysis of dextrous manipulation, assuming that there is no change of contact modes between the hand and object during a task. In contrast to conventional manipulation, we should take into consideration the manipulation system with changing kinematics and dynamics according to the change of contact modes. Therefore, we should verify the kinematical and dynamical properties of such a manipulation system with switching contact modes. Also we should execute manipulation planning taking into consideration the characteristics of the manipulation system in order to achieve our goal.

There is less work on manipulation planning. Trinkle and Hunter [10] showed global manipulation planning for quasi-static manipulation by searching feasible contact formations between each finger link and an object. However, the contacts are restricted to frictionless contacts and planning of trajectories is not discussed. Cherif and Gupta [1] discussed quasi-static manipulation planning based on cell decomposition in order to achieve dextrous manipulation by changing contact modes. However, since a search for the subgoal at each step is restricted to adjacent cells, uniform sampling of the configuration space is not guaranteed. The application of the method is limited to problems with low degrees of freedom. Yashima and Yamaguchi [11] proposed an algorithm of dynamic manipulation planning, considering the characteristics of the manipulation system by changing contact modes. The dimension of search space is reduced by utilizing object nominal trajectory and randomly sampled switching times. Desirable contact modes are determined using a cost function. The issue of the proposed algorithm depends on the specified object nominal trajectory a priori.

In work on motion planning, a new randomized motion planning algorithm based on the concepts of random trees has been shown to be valid for a uniform and rapid search of a high-dimensional state space [4], [5], [6]. For example, an algorithm known as rapidly exploring random trees (RRTs), was introduced by Lavalle and Kuffner [6]. The RRT algorithm consists of constructing a tree of feasible trajectories by extending branches toward randomly generated target points. However, these algorithm mainly deal with the problem of determining collision-free trajectories that connect a given start to the goal configuration for movable objects. Planning for manipulation by a multifingered hand is a complicated problem because of the nature of the contact such as rolling and sliding. To my knowledge, there is no study about manipulation planning from a randomized planning perspective.

In this paper, we first develop mathematical conditions that would be satisfied for manipulation planning, based on the kinematical properties of a manipulation system. These conditions give the restriction for feasible contact modes and the number of contact points. The dynamic problems for manipulation planning are not discussed here because the issues were already reported in our paper [11]. The proposed algorithm of manipulation planning is developed based on the concepts of randomized motion planning in order to explore the object configuration space, SE(3). Finally, we apply the manipulation planner to the simulation of planning for 3-D manipulation.

II. CONTACT MODES

A. Contact constraint

Fig.1 shows a manipulation system consisting of an object and a multi-fingered hand in space. We assume that each link of each finger has at most one contact point with the object, and each revolute joint has one degree of freedom. To describe the contact constraints between the object and the hand, we define a set of coordinate frames as follows: The reference frame, $\{U\}$, is fixed to the hand palm; the object frame, $\{O\}$, is fixed to the mass center of the object; the finger frame, $\{ij\}$, is fixed to the link of finger j with the *i*th contact point; The Gauss frame, $\{Oi\}$, of the object at the *i*th contact point are fixed relative to $\{O\}$. The Gauss frame, $\{Fi\}$, of the finger at the *i*th contact point is given similarly.

The relative linear velocities of $\{Oi\}$ relative to $\{Fi\}$ expressed in $\{Oi\}$ can be written as

$$\begin{bmatrix} v_{xi} \\ v_{yi} \\ v_{zi} \end{bmatrix} = \begin{bmatrix} \mathbf{G}_{xi}^T \\ \mathbf{G}_{yi}^T \\ \mathbf{G}_{zi}^T \end{bmatrix} \mathbf{V}_O - \begin{bmatrix} \mathbf{J}_{xij} \\ \mathbf{J}_{yij} \\ \mathbf{J}_{zij} \end{bmatrix} \dot{\boldsymbol{\theta}}_j$$
(1)

where $V_O \in \mathbb{R}^6$ is the linear and angular velocities of the object, and $\dot{\theta}_j \in \mathbb{R}^{n_{\theta_j}}$ is the joint velocity of the *j*th finger. n_{θ_j} indicates the number of joints of the *j*th finger. $G_{xi}^T \in \mathbb{R}^{1 \times 6}$ is the matrix which maps V_O to the *x*component of the contact velocity of the object expressed in $\{Oi\}$. $J_{xij} \in \mathbb{R}^{1 \times n_{\theta_j}}$ is the Jacobian matrix which maps $\dot{\theta}_j$ to the *x*-component of the contact velocity of the finger expressed in $\{Oi\}$. Similarly, matrices G_{yi}^T , G_{zi}^T , J_{yij} and J_{zij} are defined.

By contrast, the relative angular velocities of $\{Oi\}$ relative to $\{Fi\}$ expressed in $\{Oi\}$ can be written as

$$\begin{bmatrix} \omega_{xi} \\ \omega_{yi} \\ \omega_{zi} \end{bmatrix} = \begin{bmatrix} \widetilde{\boldsymbol{G}}_{xi}^T \\ \widetilde{\boldsymbol{G}}_{yi}^T \\ \widetilde{\boldsymbol{G}}_{zi}^T \end{bmatrix} \boldsymbol{V}_O - \begin{bmatrix} \widetilde{\boldsymbol{J}}_{xij} \\ \widetilde{\boldsymbol{J}}_{yij} \\ \widetilde{\boldsymbol{J}}_{zij} \end{bmatrix} \dot{\boldsymbol{\theta}}_j$$
(2)

where $\tilde{\boldsymbol{G}}_{xi}^{T} \in \mathbb{R}^{1 \times 6}$ is the matrix which maps \boldsymbol{V}_{O} to the *x*-component of the angular velocity of the object at the *i*th contact point expressed in $\{Oi\}$. $\tilde{\boldsymbol{J}}_{xij} \in \mathbb{R}^{1 \times n_{\theta j}}$ is the Jacobian matrix maps $\dot{\boldsymbol{\theta}}_{j}$ to the *x*-component of the angular velocity of the finger at the *i*th contact point expressed in $\{Oi\}$. Similarly, matrices $\tilde{\boldsymbol{G}}_{yi}^{T}, \tilde{\boldsymbol{G}}_{zi}^{T}, \tilde{\boldsymbol{J}}_{yij}$ and $\tilde{\boldsymbol{J}}_{zij}$ are defined.

We will consider contact between two rigid bodies as shown in Fig.2. For a rolling contact, relative motions in x_i , y_i and z_i directions are constrained by corresponding contact forces, f_{Rxi} , f_{Ryi} and f_{Rzi} which lie



Fig. 2. Contact frictional forces and relative motions

within the corresponding friction cone represented by $\sqrt{f_{Rxi}^2 + f_{Ryi}^2} \leq -\mu f_{Rzi}, f_{Rzi} \leq 0$, where μ denotes the coefficient of friction. As a result, the relative velocities, $v_{xi} = v_{yi} = v_{zi} = 0$, are required. We call the contact force model RFM (Rolling Force Model). Conversely, a sliding contact requires $v_{zi} = 0$ by a normal contact force, $f_{Szi} \leq 0$. The tangential components of the contact force lie on the boundary of the friction cone expressed by $f_{Sxi} = \tilde{\mu}_{xi} f_{Szi}, f_{Syi} = \tilde{\mu}_{yi} f_{Szi}$, where $\tilde{\mu}_{\bullet i} = \mu v_{\bullet i} / \sqrt{v_{xi}^2 + v_{yi}^2}, \bullet \in \{x, y\}$. We call the contact force model SFM (Sliding Force Model). Therefore, the contact constraints on the rolling contact and sliding contacts can be written using (1) and (2) as

$$\begin{bmatrix} \boldsymbol{G}_{Rxi}^{T} \\ \boldsymbol{G}_{Ryi}^{T} \\ \boldsymbol{G}_{Rzi}^{T} \end{bmatrix} \boldsymbol{V}_{O} = \begin{bmatrix} \boldsymbol{J}_{Rxij} \\ \boldsymbol{J}_{Ryij} \\ \boldsymbol{J}_{Rzij} \end{bmatrix} \dot{\boldsymbol{\theta}}_{j} \quad \text{for rolling (3)}$$
$$\boldsymbol{G}_{Szi}^{T} \boldsymbol{V}_{O} = \boldsymbol{J}_{Szij} \dot{\boldsymbol{\theta}}_{j} \quad \text{for sliding (4)}$$

where subscripts R and S denote rolling and sliding contacts.

Summing (3) and (4) for all contacts yields

$$\boldsymbol{G}_{A}^{T}\boldsymbol{V}_{O} = \boldsymbol{J}_{A}\dot{\boldsymbol{\theta}} \tag{5}$$

where there are n_C contacts, consisting of n_R rolling contacts and n_S sliding contacts. n_{θ} and n_F denote the sum of joints and fingers, respectively.

Similarly, matrices G_{Ry} , G_{Rz} , G_{Sz} , J_{Ry} , J_{Rz} and J_{Sz} are defined.

When the object velocity, V_O , is uniquely determined for a given joint velocity, $\dot{\theta}$, the object motion can be restricted at each instant by the hand motion. A manipulation system satisfying this property is said to be kinematically determinate. Otherwise, there are infinite solutions for (5), and the object cannot be manipulated by the hand motion. From (5), the conditions for kinematically determinateness can be written as

 $3n_R + n_S \ge 6$, $n_R + n_S = n_C$, G_A is full rank (6)

B. Feasible contact modes

When a manipulation system is kinematically determinate, $(3n_R + n_S)$ relative motions of contact points are restricted by constraint forces, and the motion of the object is uniquely determined by the motion of the hand. In this case, we can not only manipulate the object along the desired object trajectory by the hand, but we can plan the motion of the hand so that more than $(3n_R + n_S)$ relative motions of contact points could be zero using the redundant degrees of freedom of the hand. Based on this idea, we can plan more contact modes besides rolling and sliding contact modes. Consequently, we can realize dextrous manipulation. In this section, we discuss the feasible contact modes generated by manipulation planning and their conditions.

The feasible contact modes generated by manipulation planning can be classified into eight contact modes, as shown in Table.I. The \times indicates that the component of relative velocities is zero. The n_{Pi} shows that sum of the component of zero-relative velocities. No.1~4 in Table.I belong to a class of the SFM. By increasing the component of zero-relative velocities based on planning, we can plan new contact modes such as "Sliding with Roll, Sliding with Spin and Pure Sliding" from "Sliding" contact modes generated by constraint forces. On the other hand, no.5~8 belong to a class of the RFM. Similarly, we can plan new contact modes such as "Pure Rolling, Pure Spin and Stuck" from "Rolling" contact modes generated by constraint forces. Fig.3 illustrates motion of each feasible contact mode.

Combining (1) and (2) yields

$$\begin{bmatrix} v_{xi} & v_{yi} & v_{zi} & \omega_{xi} & \omega_{yi} & \omega_{zi} \end{bmatrix}^T = \\ \begin{bmatrix} G_{xi} & G_{yi} & G_{zi} & \widetilde{G}_{xi} & \widetilde{G}_{yi} & \widetilde{G}_{zi} \end{bmatrix}^T V_O \\ -\begin{bmatrix} J_{xi}^T & J_{yi}^T & J_{zi}^T & \widetilde{J}_{xi}^T & \widetilde{J}_{yi}^T & \widetilde{J}_{zi}^T \end{bmatrix}^T \dot{\boldsymbol{\theta}}_j$$
(7)

The kinematic equation for each feasible contact mode shown in Table.I can be derived by multiplying (7) by a contact mode selection matrix $B_i \in \Re^{n_{P_i} \times 6}$ as :

$$\boldsymbol{H}_{Oi}^{T}\boldsymbol{V}_{O} = \boldsymbol{H}_{Fij}\dot{\boldsymbol{\theta}}_{j} \tag{8}$$

TABLE I Feasible contact modes

| No. | Contact mode | Force model | v_{xi} | v_{yi} | v_{zi} | ω_{xi} | ω_{yi} | ω_{zi} | n_{Pi} |
|-----|-----------------|----------------|----------|----------|----------|---------------|---------------|---------------|----------|
| 1 | Sliding | SFM | 0 | 0 | × | 0 | 0 | 0 | 1 |
| 2 | Roll sliding | SFM | 0 | 0 | × | 0 | 0 | × | 2 |
| 3 | Spin sliding | SFM | 0 | 0 | × | × | × | 0 | 3 |
| 4 | Pure sliding | SFM | 0 | 0 | × | × | × | × | 4 |
| 5 | Rolling | RFM | × | × | × | 0 | 0 | 0 | 3 |
| 6 | Pure rolling | RFM | × | × | × | 0 | 0 | × | 4 |
| 7 | Pure spin | RFM | × | × | × | × | × | 0 | 5 |
| 8 | Stuck | RFM | X | X | X | Х | Х | X | 6 |



Fig. 3. Motion of feasible contact modes

where

$$\boldsymbol{H}_{Oi}^{T} = \boldsymbol{B}_{i} [\boldsymbol{G}_{xi} \ \boldsymbol{G}_{yi} \ \boldsymbol{G}_{zi} \ \widetilde{\boldsymbol{G}}_{xi} \ \widetilde{\boldsymbol{G}}_{yi} \ \widetilde{\boldsymbol{G}}_{zi}]^{T}$$

$$\boldsymbol{H}_{Fij} = \boldsymbol{B}_{i} [\boldsymbol{J}_{xi}^{T} \ \boldsymbol{J}_{yi}^{T} \ \boldsymbol{J}_{zi}^{T} \ \widetilde{\boldsymbol{J}}_{xi}^{T} \ \widetilde{\boldsymbol{J}}_{yi}^{T} \ \widetilde{\boldsymbol{J}}_{zi}^{T}]^{T}$$

Summing (8) for all contacts yields

$$\boldsymbol{H}_{O}^{T}\boldsymbol{V}_{O} = \boldsymbol{H}_{F}\dot{\boldsymbol{\theta}}$$
(9)

where $\boldsymbol{H}_{O}^{T} \in \Re^{n_{P} \times 6}$, $\boldsymbol{H}_{F} \in \Re^{n_{P} \times n_{\theta}}$ and $n_{P} = \sum_{i}^{n_{C}} n_{Pi}$. A manipulation system is said to be kinematically manipulable when we can find a joint velocity, $\dot{\boldsymbol{\theta}}$, which can accommodate an arbitrary object velocity, V_{O} , for an assigned contact mode. From (9), the conditions for kinematically manipulability can be written as

$$3n_R + n_S \le n_P \le n_\theta$$
, \boldsymbol{H}_F is full rank (10)

These conditions for kinematically determinateness and manipulability in (6) and (10) are used to decide the feasibility of manipulation in manipulation planning in the next section.

III. RANDOMIZED MANIPULATION PLANNING

A. Global planning level

We discuss manipulation planning for three-dimensional manipulation by switching contact modes, based on the idea of random motion planning. In this paper, the aim of manipulation planning is to find feasible trajectories of motions and driving torques for the multi-fingered hand to manipulate the object from an initial configuration, $X_{init}(R_{Oinit}, x_{Oinit}) \in SE(3)$, to a specified final

desired configuration, $X_{goal}(R_{Ogoal}, x_{Ogoal}) \in SE(3)$. We denote by $R_O \in SO(3)$ and $x_O \in \Re^3$ the orientation and the position of $\{O\}$ relative to $\{U\}$ respectively. This problem, then, corresponds to an inverse problem.

The proposed algorithm of manipulation planning has a two-level planning scheme consisting of a global planning level and a local planning level. The global planner explores the object configuration space and constructs random trees extending to X_{goal} . By contrast, the local planner checks if feasible trajectories (trees) exist that connect subgoals (nodes). In this section, we present the algorithm of a global planner.

In the global planning level, it is necessary to explore the object configuration space, SE(3), rapidly and uniformly. Based on the ideas of random motion planning using the concepts of random trees, introduced by LaValle et.al. [6], we propose a novel algorithm for manipulation planning which explores a configuration space, SE(3). Fig.4 illustrates exploring random trees in the object configuration space. The planner picks the candidate for subgoal, $X_{rand} \in SE(3)$, at random in the object configuration space, then selects the nearest subgoal, X_{near} , from X_{rand} from the existing random trees. Using a local planner to connect pairs of subgoals, X_{near} and X_{rand} , the random trees are expanded in the configuration space. If the connection is not feasible, a new candidate, X_{rand} , is re-selected at random. If the connection is feasible, the local planner tries again to connect X_{rand} to X_{aoal} . Also when the connection is feasible, the search is terminated. Otherwise, this process is performed iteratively until the random trees extend to X_{goal} . Note that the forbidden region in Fig.4 is treated differently from an obstacle within a path planning problem since the forbidden region is not fixed and unknown in the configuration space.

The EXTEND_RANDOM_TREE function in Fig.5 shows the algorithm of the global planner. First, the REACHABLE function checks if the local planner can connect X_{init} to X_{goal} directly. Unless there is such a trajectory connecting X_{init} to X_{goal} , expanding random trees from X_{init} is executed iteratively. At each step, the RANDOM_CONFIG function selects a candidate for the object subgoal, X_{rand} , at random in the object configuration space. In order for subgoals to be distributed uniformly over the configuration space, the restriction of the range of exploration in the configuration is imposed.

The NEAREST_NEIGHBOR function chooses the nearest subgoal, X_{near} , from X_{rand} according to a weighted Euclidean distance for position and orientation. Then, the function REACHABLE(X_{near}, X_{rand}) checks whether random trees can be expanded from X_{near} to X_{rand} . If the random tree cannot reach from X_{near} to X_{rand} , then we go back to the second step. Otherwise, the ADD_NEW_SUBGOAL function adds X_{rand} to the random trees as a new subgoal. In addition,



Fig. 4. Exploring random tree in object configuration space

| EXTEND_RANDOM_TREE | | | | | |
|--------------------|--|--|--|--|--|
| 1 | if (REACHABLE(X_{init}, X_{goal})=TRUE) then STOP | | | | |
| 2 | $X_{rand} \longleftarrow 	ext{RANDOM_CONFIG}$ | | | | |
| 3 | $X_{near} \longleftarrow \text{NEAREST_NEIGHBOR}(X_{rand})$ | | | | |
| 4 | if (REACHABLE(X_{near}, X_{rand})=TRUE) then | | | | |
| 5 | ADD_NEW_SUBGOAL(X_{rand}) | | | | |
| 6 | if (REACHABLE(X_{rand}, X_{goal})=TRUE) then STOP | | | | |
| 7 | else goto the 2 nd step | | | | |
| 8 | else goto the 2 nd step | | | | |
| | | | | | |



REACHABLE(X_{rand}, X_{goal}) checks if random trees can be expanded from X_{rand} to X_{goal} . If so, the exploration is terminated. Otherwise, we go back to the second step. The EXTEND_RANDOM_TREE is terminated if the iteration of this function exceeds a specified upper limit.

B. Local planning level

In the local planning level, contact modes and an object trajectory connecting between a specified two object configuration, $X_A(\mathbf{R}_A, \mathbf{p}_A)$ and $X_B(\mathbf{R}_B, \mathbf{p}_B) \in SE(3)$, in a switching time interval, ΔT , are first assigned. Then the inverse problem is solved at each instant for the given object trajectory in order to find feasible joint driving torque trajectories which would satisfy all manipulation constraints.

The REACHABLE(X_A , X_B) function in Fig.6 shows the algorithm of a local planner. First, the GENERATE_TRAJ(X_A , X_B) function generates the object trajectory which connects X_A with X_B . Cartesian position trajectory, $x_O(t) \in \Re^3$, is given by a cubic polynomial which satisfies the following boundary conditions

$$\boldsymbol{x}_O(0) = \boldsymbol{p}_A, \quad \boldsymbol{x}_O(\Delta T) = \boldsymbol{p}_B, \quad \dot{\boldsymbol{x}}_O(0) = \dot{\boldsymbol{x}}_O(\Delta T) = 0$$

On the other hand, The relative changes of the orientation between X_A and X_B can be represented by a rotation about an equivalent axis, $\hat{k}_E \in \Re^3$, by an equivalent angle, $\phi_E \in \Re^1$. Therefore, Cartesian orientation trajectory can be expressed as the variation of rotation angle, $\phi(t) \in \Re^1$, from 0 to ϕ_E about the rotation axis, \hat{k}_E . The orientation





trajectory, $\phi(t)$, is also given by a cubic polynomial which satisfies the following boundary conditions

$$\phi(0) = 0, \quad \phi(\Delta T) = \phi_E, \quad \dot{\phi}(0) = \dot{\phi}(\Delta T) = 0$$

The orientation of X_B relative to X_A can be written as $R_E = R_A^T R_B$. Therefore, ϕ_E and \hat{k}_E are given by [8]

$$\phi_E = \cos^{-1}\left(\frac{\operatorname{trace}(\boldsymbol{R}_E) - 1}{2}\right) \qquad (11)$$

$$\widehat{\boldsymbol{K}}_{E} = \frac{1}{2\sin\phi_{E}} (\boldsymbol{R}_{E} - \boldsymbol{R}_{E}^{T})$$
(12)

where $\widehat{\mathbf{K}}_E \in \mathbb{R}^{3\times 3}$ is a skew-symmetric matrix for $\widehat{\mathbf{k}}_E$. Then, an object angular velocity vector, $\boldsymbol{\omega}_O(t) \in \mathbb{R}^3$, can be written using the object angular velocity, $\dot{\phi}(t)$, as

$$\boldsymbol{\omega}_O(t) = \dot{\phi}(t) \boldsymbol{R}_A \hat{\boldsymbol{k}}_E \tag{13}$$

The RANDOM_CM function chooses a contact mode at random which satisfies (6) and (10). The INVERSE_PROBLEM function constructs the kinematic and dynamic equations for the selected contact mode. When the inverse dynamic problem has the solution of the joint torques for a given object trajectory, the REACHABLE function returns TRUE. Otherwise, it returns FALSE. The details of solution method for the inverse dynamic problem are shown in our paper [11].

Next, we discuss the relationship between the switching of contact mode and reachability. As shown at the top of Fig.7, when the forbidden region is too large or subgoals are constructed in narrow configuration space surrounded by the forbidden region (e.g. X_i in this figure), it is difficult to expand to X_{goal} even if the candidate for subgoal is selected at random in the configuration space. We should notice that the kinematics and dynamics of the manipulation system varies according the change of the contact mode. Therefore, changing contact modes makes the forbidden region vary and assist the expansion of the random trees from the stuck subgoal, as shown at the bottom of Fig.7. In other words, the switching contact modes contributes to the enhancement of reachability.

IV. COMPUTER SIMULATION

In this section, we verify the effectiveness of the proposed manipulation planning by considering the manipulation system, which consists of one elliptic object and a three 3-degrees of freedom fingered hand with a flat



Fig. 7. Change of forbidden regions by switching contact modes



Fig. 8. Simulation model

rectangular effector on the third link, as shown in Fig.8. Each first joint of the three fingers locates in the same position, and is independently actuated of the other. Each flat link has the contacts with the object. The radiuses of the object is 0.2, 0.2 and 0.16 m. The mass of the object is 1.0Kg. The first and second links are 0.14 and 0.2 m long respectively. The dimensions of the flat link is 0.2 m by 0.07 m. The mass of each link is 0.5 Kg. The coefficient of friction is 0.5.

In order for the manipulation system to be kinematically manipulable and determinate, the following constraints about the number of rolling and sliding contacts are imposed by (6) and (10):

$$6 \le 3n_R + n_S \le n_P \le 9 \tag{14}$$

Therefore, the eight feasible contact modes shown in Table.II can be considered. Each feasible contact mode consists of the combinations of contacts mode shown in Table.I. No.1~7 correspond to the case of $n_R = 2$, $n_S = 1$. While no.8 corresponds to the case of $n_R = 3$, $n_S = 0$. Because each feasible contact mode in Table.II should be considered for each of three contact points, the number of candidates for the feasible contact modes amounts to 31 combinations. The feasible mode is determined among the combinations in the REACHABLE function.

We consider a manipulation task with rotation (the equivalent axis, $\hat{k} = [-0.5, 0.5, 0.707]^T$, and the equiv-

TABLE II COMBINATIONS OF FEASIBLE CONTACT MODES



Fig. 9. Exploring random trees for object position and orientation.

alent angle, $\phi = 30$ deg) and translation (0.02, 0.02, 0.02) m in x, y and z axis directions) from the initial object configuration. The fixed $\Delta T=1.0$ sec is used. The algorithm was implemented in MATLAB on 1.4 GHz Intel Pentium IV PC with 256 MB of memory running Windows 2000. Fig.9 (a) shows the explored subgoals of the object position (x_O, y_O, z_O) with respect to the base frame. The numbers attached to each subgoal indicate the sequence of the generated subgoals and the contact mode numbers for each of three contact points. The bold lines show the trees which extended to the X_{qoal} . Similarly, Fig.9 (b) shows the explored subgoals of the object orientation. The orientation is expressed by a Gauss map for the unit vectors of the principal axes of frame, $\{O\}$, of each subgoal. The initial orientation agrees with the x, y and z axes in Fig.9 (b). After a total of 399 subgoal candidates were selected at random, 29 subgoals were locally reachable among them. Finally, we got the solution of trajectory with five switching contact modes. Fig.10 shows the snapshot for the solution trajectory.

V. CONCLUSION

We have presented randomized manipulation planning with switching contact modes. When a manipulation system is kinematically determinate and manipulable, we can not only manipulate an object along the desired trajectory of the object by the motion of the multi-fingered hand, but also can extend a class of feasible contact modes by using the redundant degrees of freedom of the multifingered hand. In addition, switching contact modes can improve reachability for manipulation planning. These ideas contribute to dextrous manipulation. We proposed the new algorithm for manipulation planning based on the concepts of random trees. We showed the validity of



the planner by simulating manipulation planning for 3-D manipulation.

VI. REFERENCES

- M.J. Cherif, K.K. Gupta, "Planning quasi-static fingertip manipulations for reconfiguring objects", *IEEE Transactions on Robotics and Automation*, Vol. 15, no. 5, 1999, pp.837-848.
- [2] A.A. Cole, J.E. Hauser and S.S. Sastry, "Kinematics and control of multifingered hands with rolling contact", *IEEE Transactions on Aotomatical Control*, Vol. 34, no. 4, 1989, pp.398-404.
- [3] A.A. Cole, P. Hsu and S.S. Sastry, "Dynamic control of sliding by robot hands for regrasping", *IEEE Transactions* on Robotics and Automation, Vol. 8, no. 1, 1992, pp.42-52.
- [4] E. Frazzoli, M.A. Dahleh and E. Feron, "Real-time motion planning for agile autonomous vehicles", *Journal* of Guidance, Control and Dynamics, Vol. 25, no. 1, 2002, pp.116-129.
- [5] R. Kindel, D. Hsu, J.C. Latombe, S. Rock, "Kinodynamic motion planning amidst moving obstacles", *In Proc. of IEEE Int. Conf. on Robotics and Automation*, 2000.
- [6] S.M. LaValle, J.J. Kuffner, Jr, "Randomized kinodynamic planning", *The International Journal of Robotics and Research*, Vol. 15, no. 5, 2001, pp.378-400.
- [7] D.J.Montana, "The kinematics of contact and grasp", Int. Journal of Robotics Research, Vol. 7, no. 3, 1988, pp.17-32.
- [8] R.M. Murray, Z. Li and S.S. Sastry, A Mathematical Introduction to Robotic Manipulation. CRC Press, 1993.
- [9] N. Sarker, X. Yun and V. Kumar, "Dynamic control of 3-D rolling contacts in two-Arm manipulation", *IEEE Transactions on Robotics and Automation*, Vol. 13, no. 3, 1997, pp.364–376.
- [10] J.C. Trinkle, J.J. Hunter, "A framework for planning dextrous manipulation", In Proc. of IEEE Int. Conf. on Robotics and Automation, 1991.
- [11] M. Yashima and H. Yamaguchi, "Dynamic motion planning whole arm grasp systems based on switching contact modes", In Proc. of IEEE Int. Conf. on Robotics and Automation, 2002.