Pivoting Based Manipulation by Humanoids: a Controllability Analysis

Eiichi Yoshida, Mathieu Poirier, Jean-Paul Laumond, Rachid Alami and Kazuhito Yokoi

Abstract— **Pivoting manipulation has such advantages as dexterity and safety over other methods to move bulky or heavy objects. In this paper we aim to show that a polyhedral object can be displaced to arbitrary position and orientation on a plane (i.e. such a pivoting system is controllable). More than that we show it is small time controllable, i.e. the reachable space from a starting point contains always a neighbor no matter how cluttered the environment is. As a consequence of this analysis, we propose a steering method to plan a manipulation path to be performed by a humanoid robot: first we use a classical nonholonomic path planner that accounts for the robot motion constraints, and then we transform that path into a sequence of pivoting operations. While the feasibility of elementary pivoting tasks has been already experienced by the humanoid robot HRP-2 [2], we present here the very first simulations of the plans generated by our steering method.**

I. INTRODUCTION

Humanoid robots have a high mobility and adaptability to human environments thanks to its legged locomotion and whole-body motion capacity. They are now expected to be used in various application areas to assist or substitute humans. One of the important issues to improve their capacity is manipulation of various objects. In this paper we focus on a whole-body manipulation by a humanoid robot to move bulky objects that are difficult to be dealt with other types of robots. We adopt "pivoting" manipulation introduced by Aiyama et al. for this purpose [1], since we can expect more precise positioning, stability and adaptability over other methods like pushing or lifting. As shown in Fig. 1, we have successfully implemented a pivoting manipulation using a humanoid robot [2].

There have been studies on motion planning for other types of manipulation, like pushing [3], [4] or tumbling [5], to move the object to the desired position by introducing sophisticated control methods.

However, to our knowledge no planning algorithms have been developed to displace an object to an arbitrary goal position just by pivoting. The contribution of this paper is to propose a modeling and an analysis of the problem. Such an analysis leads to a well-grounded planning algorithm allowing a humanoid robot to move a cumbersome polyhedral object to any desired position by pivoting it around its corners.
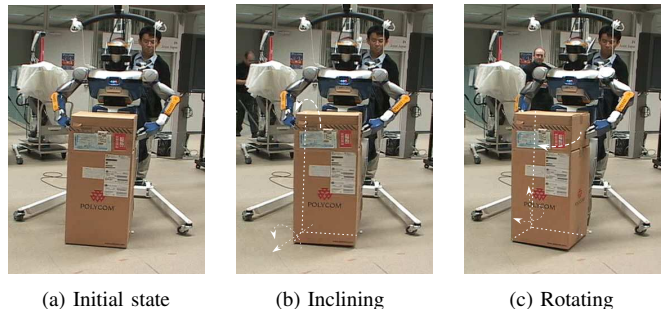
(a) Initial state   (b) Inclining   (c) Rotating

Fig. 1. Experiment of pivoting motion

The method and its completeness analysis are based on the controllability study of the considered system. We show in section IV that certain sequences of elementary pivoting tasks verify some topological property. The definition of the so-called topological property is introduced in section II: it accounts for the fact that the reachable space from any given starting point contains always a neighbor of such starting point no matter how cluttered the environment is. As a consequence of that property, it is possible to approximate any collision-free path by a sequence respecting both the obstacle avoidance and the pivoting constraints.

In a second stage we address the constraints imposed to the robot. First the robot should walk while pivoting. This means it should face the object to be moved. Such a system is constrained to move in the direction imposed by the body direction of the robot. It is desirable for the robot to move forward than sideways (see Fig. 1) from the viewpoint of ease of manipulation and stability: this is a classical nonholonomic constraint well known in mobile robotics [6]. Then the steering method consists in combining both a steering method used for a nonholonomic car-like robot together with elementary pivoting motion sequences.

The validity of proposed steering method is verified by simulations where the motion of the manipulation box and the humanoid robot are planned in section V. The method is proven to verify a topological property that allows the planner to be plugged into a general collision-free motion planning scheme. The controllability analysis results as well as the planning scheme introduced in this paper are finally put in perspective in the general framework of manipulation task planning in section VI.

II. SMALL-TIME CONTROLLABILITY, STEERING METHODS AND THEIR TOPOLOGICAL PROPERTY

According to the control literature (i.e. [7]), a system is said to be controllable if it may reach any configuration from
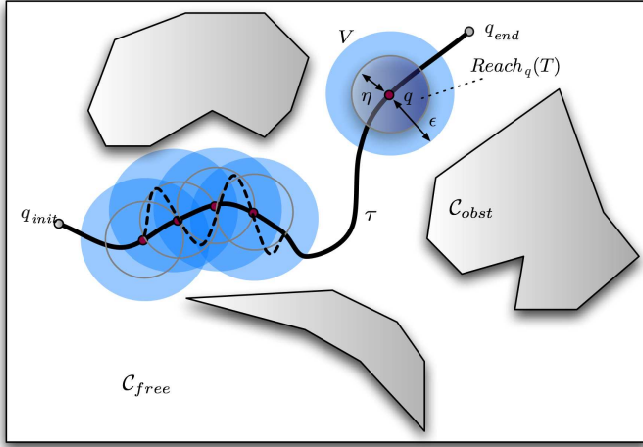
Fig. 2. A system is small-time controllable from q if $Reach_q(T)$ contains a neighborhood of q for all neighborhoods V for any time $T > 0$.

any other one. It is said to be small-time controllable at q if the set of configurations $Reach_q(T)$ that can be reached from a configuration q before a given time $T(>0)$ contains a neighborhood of q (of course such a property should hold for any T). In geometric words, as shown in the top-right of Fig. 2, the system can move anywhere in the area η without leaving an imposed neighborhood V. A system is said to be small-time controllable if it is at any configuration q. Small-time controllability is central in path planning. Indeed the main consequence is depicted in Fig. 2: any collision-free path (not necessarily admissible by the system) can be approximated by a sequence of both collision-free and admissible ones. This type of analysis plays a critical role in nonholonomic motion planning [6].

Small-time controllability is a property of the considered system. It can be proven by applying the so-called Lie Algebra Rank Condition (LARC) [7]. Let us now consider the problem to steer a system. We call "steering method" a method that computes an admissible path from a starting configuration to a goal (in the absence of obstacle). Devising a steering method is not an easy task. For instance we know that a mobile robot with trailers is controllable; however the problem of steering it (when the trailer hocking-up point is not centered on the middle of the previous trailer) is still an open problem after more than twenty years of study. Such a problem is critical for motion planning. It is even more difficult if we want to account for small-time controllability. A system may be proved to be small-time controllable, devising steering methods that account for small-time controllability is another story (see [6], [9], [8] for a detailed review of the concepts). We have introduced the notion of *topological property* in [10] to characterize the steering methods that accounts for small-time controllability. More precisely, let us consider a small-time controllable system. A steering method ST for that system verifies the topological property (TP) if and only if, for any neighborhood V and any configuration q, the set of configurations reachable from q by a path computed with ST contains a neighborhood of q.

III. PIVOTING AND SMALL-TIME CONTROLLABILITY

This section introduces the pivoting model we are considering and shows that it is small-time controllable. Based on LARC, the system is small-time controllable if the vector fields of the motion and their Lie Brackets are linearly independent. The robot is assumed to be able to maintain the balance of the polyhedron on a single edge. Then the robot uses both corners of the edge to generate alternate sequence of pivoting motions. Therefore planning a pivoting sequence appears as a problem of displacing a line segment AB on a plane by rotating itself alternatively at one of its extremity points A and B. Let us write the moving segment as a control system. Let $2l$ denote the length of the line segment. The configuration of the line segment is expressed as that of its center $O(x, y)$ and its orientation θ. At each step of the pivoting sequence, two elementary rotating motions are allowed for the segment.

Let us consider the left (counterclockwise) rotation of angle ε around the point A (Fig. 3). Assuming that the starting configuration is $(0, 0, 0)$, the final configuration is $(l(\cos \varepsilon - 1), l \sin \varepsilon, \varepsilon)$. The vector field L for left rotation turn corresponding to this motion is (see Appendix for detailed derivation):

$$L = \begin{pmatrix} -l \sin \theta \\ l \cos \theta \\ 1 \end{pmatrix}. \tag{1}$$

It can be understood as the tangent velocity of the point O at the configuration $q(x, y, \theta)$. The same computation applies for right turn; turning around the point B corresponds to a counterclockwise motion with associated vector field

$$R = \begin{pmatrix} l \sin \theta \\ -l \cos \theta \\ 1 \end{pmatrix}. \tag{2}$$

The Lie bracket of both L and R is computed as follows (See Appendix for the derivation):

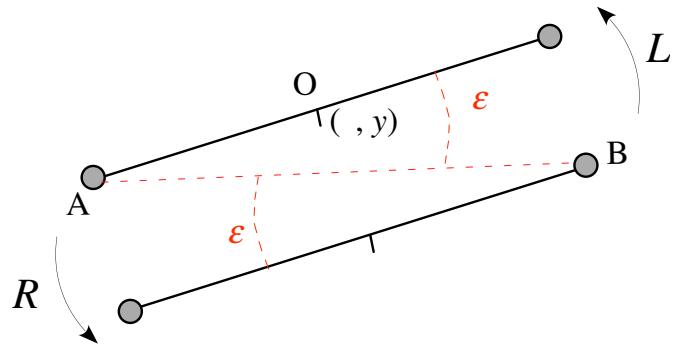$$[L, R] = \begin{pmatrix} -2l \cos \theta \\ -2l \sin \theta \\ 0 \end{pmatrix}. \tag{3}$$

Fig. 3. Pivoting problem: displacing a line segment A or B

The three vector fields L, R and $[L, R]$ are linearly independent. Then the LARC condition holds. The pivoting polyhedron is a small-time controllable system. The reachable space from any starting configuration contains always a neighbor no matter how cluttered the environment is.

IV. IMPLEMENTATION OF PIVOTING STEERING METHOD

Now the question is to devise a steering method that accounts for small-time controllability, i.e. that verifies the topological property TP.

A. A manipulation sequence that verifies TP

Let us consider the sequence of three pivoting motions illustrated in Figure Fig. 4. Here the starting configuration is supposed to be $(0,0,0)$. The angles of the three rotations are denoted by α_1, α_2 and α_3 respectively. We can compute the final configuration as:

$$
\begin{aligned}
q_f &= f(\alpha_1, \alpha_2, \alpha_3) \qquad (4)\\
&= \begin{cases}
x = -l + 2l\cos\alpha_1 - 2l\cos(\alpha_1 + \alpha_2) \\
\quad + l\cos(\alpha_1 + \alpha_2 + \alpha_3), \\
y = 2l\sin\alpha_1 - 2l\sin(\alpha_1 + \alpha_2) \\
\quad + l\sin(\alpha_1 + \alpha_2 + \alpha_3), \\
\theta = \alpha_1 + \alpha_2 + \alpha_3.
\end{cases}
\end{aligned}
$$

Assuming that each α_i belongs to $[-\pi/4, \pi/4]$, let us derive the inverse function f^{-1} of f:

$$
f^{-1}(x, y, \theta) = \begin{cases}
\alpha_2 = \pm\arccos(\dfrac{X^2 + Y^2 - 2}{-2}), \\
\alpha_1 = \phi \pm \arccos(\dfrac{X}{\rho}), \qquad (5) \\
\alpha_3 = \theta - \alpha_1 - \alpha_2.
\end{cases}
$$

where

$$
\begin{aligned}
X &= \frac{x + l(1 - \cos\theta)}{2l}, \\
Y &= \frac{y - l\sin\theta}{2l}, \\
\rho &= \sqrt{(1 - \cos\alpha_2)^2 + \sin\alpha_2{}^2}. \qquad (6)
\end{aligned}
$$

Since a square of both sides of equation is employed in the course of derivation, it is necessary to choose correctly the sign of variables. Note also that the above calculation is
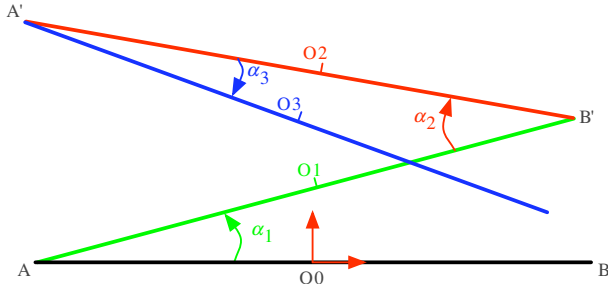
applicable because we assume that the argument of $\arccos$ stays within $[-1, 1]$. This corresponds to an inverse kinematics of a planar manipulator arm with three rotation joints.

Even though such a sequence of three pivoting motions does not allow the system to reach a configuration far from the origin, it is important to know if it can reach any configuration in a sufficiently small neighbor of the origin.

Let us consider the following neighbor V_ε around the origin $(0,0,0)$: $V_\varepsilon = D_\varepsilon \times (-\varepsilon, \varepsilon)$ as shown in Fig. 5, where D_ε shows the disk for x-y coordinates whose radius is ε. The total length of the path is $l(|\alpha_1| + |\alpha_2| + |\alpha_3|)$. It then suffices to choose all the three α_i smaller than $\varepsilon/3l$ to guarantee that the path will not escape D_ε. Considering the variable θ, the total variation is bounded by $|\alpha_1| + |\alpha_2| + |\alpha_3|$. Therefore choosing all the α_i smaller than $\min(\varepsilon/3l, \varepsilon/3)$ is a guarantee for not escaping V_ε. $\square$

B. The steering method

Now we address the problem of how to build a steering method that allows the system to reach *any* configuration, even far from the origin.

At this stage we consider the constraints imposed by the robot. When the object is manipulated by a whole-body humanoid robot motion as shown in Fig. 1, the heading direction of the object is constrained by the robot posture. Considering the stability, arm reachability and visibility, the humanoid robot can manipulate the object more easily in forward and backward direction than along sideway directions. It is therefore reasonable to plan the object pivoting motion in such a way that the object heading direction coincides with front-side of humanoid robot. This implies that the humanoid robot behaves as a mobile robot that cannot move sideway. Then we may benefit from the steering methods for mobile robots. Among all of them, the simplest one is based on the computation of the so-called Reeds and Shepp paths [12]. The Reeds and Shepp paths are the optimal length paths for a car-like robot moving both forward and backward with a lower bounded turning radius. Reeds and Shepp paths are sequence of straight-line segments and arcs of a circle (with minimum radius). Because they are optimal paths, the Reeds and Shepp paths are a way to define a steering method verifying TP for car-like robots [6].

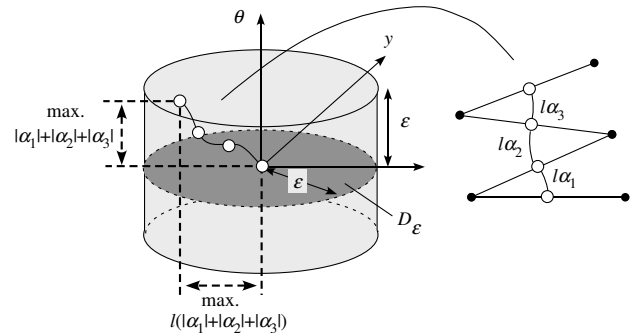Fig. 4. A sequence of three pivoting operations to move to an arbitrary neighborhood position.

Fig. 5. Analysis on small-time controllability of three-rotation pivoting steering method. The pivoting maneuver is bounded by $l(|\alpha_1| + |\alpha_2| + |\alpha_3|)$ for x-y coordinates and $|\alpha_1| + |\alpha_2| + |\alpha_3|$ for θ.

Then the simple idea to devise a steering method for our humanoid robot moving a polyhedron by pivoting is to approximate Reeds and Shepp paths by the elementary sequences of three pivoting motions introduced above. First the algorithm checks whether the endpoints of the Reeds and Shepp path are connectable by a simple sequence of three pivoting motions. If it is not possible, then the Reeds and Shepp path is split into two parts and the algorithm attempts to apply the three-rotation pivoting. This dichotomy procedure is repeated until all the components of Reeds and Shepp path is converted into a sequence of pivoting motions. Because of the topological property verified by the elementary pivoting sequences, the dichotomy algorithm converges in finite time. Moreover the resulting steering method inherits from the TP of Reeds and Shepp paths.

V. SIMULATION RESULTS

The proposed algorithm is implemented and simulated on a motion planning software kit KineoWorks[TM] [14]. To verify the effectiveness of pivoting steering method, only the motion of the object is considered in an environment without obstacles. The size of the manipulated object is 0.8m, 0.6m and 1.3m in width, depth and height respectively. The radius for Reeds and Shepp steering method is 2m and only one of the edges in the bottom face is used during manipulation. The maximum rotation angle α_{max} for pivoting is set to 22.5 degrees.

In the simulation, the planning algorithm is applied to two cases, from the initial position and orientation of the object $q_i(x_0, y_0, \theta_0) = (0, 0, 0 \text{ deg})$, to the goals $q_f(0.5, 3, 0 \text{ deg})$. Fig. 6 shows the pivoting sequence (solid line) and the Reeds and Shepp path (dotted line). The pivot points where the object is supported during rotation is designated by round dots.

In the figure we can verify that the proposed algorithm allows the object to reach exactly to the goal configurations. As can be seen in Fig. 6, the planned path has large lateral displacement; we can observe that the object is transferred to the goal configuration by using switch-back operations.

A motion generation method for HRP-2 humanoid robot to perform the planned pivoting movements is under development. The robot holds the object by two arms as shown in Fig. 1 to perform the pivot motion generated the proposed planning method.

Fig. 7 shows the snapshots of the simulated pivoting manipulation for the case of $q_f(3, 1.5, 0 \text{ deg})$. The waist of the robot is planned so that its heading direction coincides with the orientation of Reeds and Shepp path. The upper body of the robot is calculated using analytical arm inverse kinematics from the fixed position and orientation of the hands on the box. The yaw-axis angle is synchronized to the object yaw rotation. Leg motion is currently not implemented yet.

As can be seen the robot can generate the desired box pivoting motion using upper body and heading direction provided by Reeds and Shepp curve. This result shows the
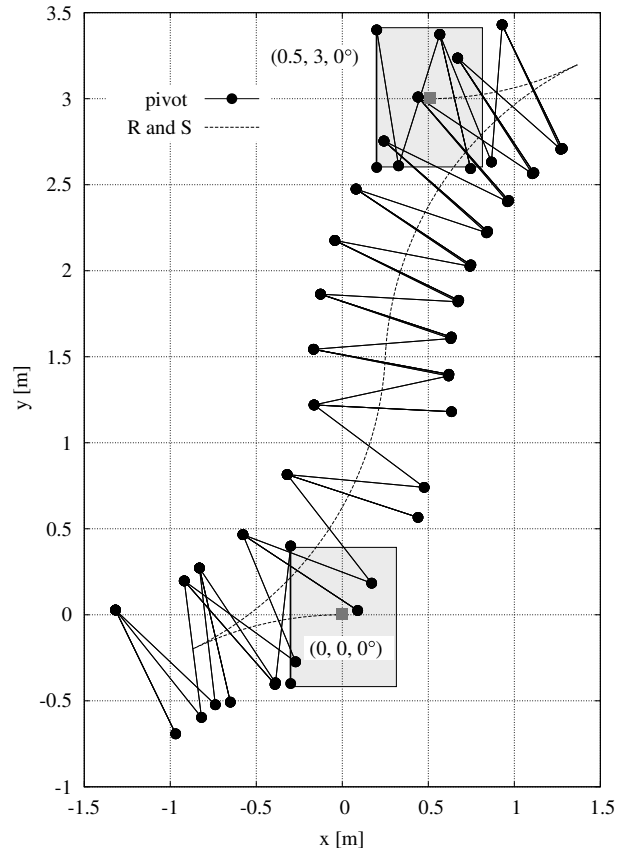

Fig. 6. Planned result for pivoting motion to $q_f(0.5, 3, 0 \text{ deg})$. In this case a back and forth motion is planned since there is a large motion in lateral direction. A feasible pivoting sequence is also derived.

proposed pivot planning method is feasible enough to be implemented for the robot by integrating leg motion.

VI. PERSPECTIVES

So far the pivoting steering method between any configurations has been well established. We are now ready to integrate it motion planning techniques.

Recent improvements in computational capacity have made it possible for sampling-based methods to solve complicated motion planning problems. A sampling-based method consists in searching a collision-free path in a roadmap generated as a graph whose nodes are randomly sampled in the free space $\mathcal{C}_{free}$ in configuration space $\mathcal{C}$. Typical sampling-based methods are known as RRT (Rapidly-exploring random tree) [15] or PRM (Probabilistic RoadMap)[16].

As stated in section II, because the pivoting sequence is small-time controllable, once a collision-free path is found it can be straightfowardly transformed to the path by the steering method.

In our case, if a collision-free Reeds and Shepp path in $\mathcal{C}_{free}$ composed of line segments and arcs of circles is planned, it can be transformed into a sequence of pivoting manipulations. When the clearance between obstacles and the manipulated object is large enough, we can use the maximum displace $\Delta q_{max}(\Delta x_{max}, \Delta y_{max}, \Delta \theta_{max})$ to transform
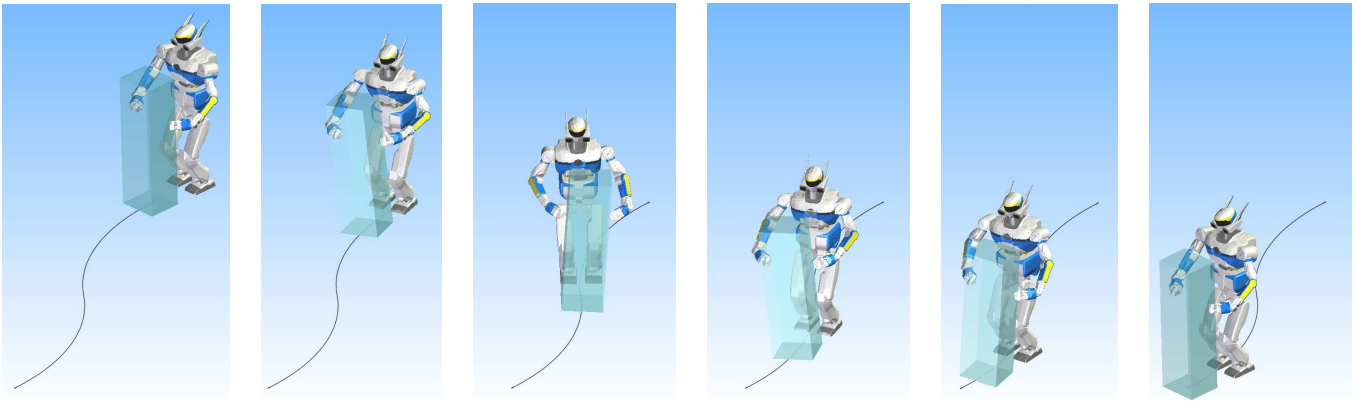
Fig. 7. Planning result of pivoting with HRP-2 humaniod robot holding the box to $q_f(3.0, 1.5, 0 \deg)$, where arm configurations are calculated using inverse kinematics from fixed contact points on the box. The figure also shows the Reeds and Shepp path whose orientation is the same as the heading direction of the robot.

the Reeds and Shepp path into pivoting sequence of smaller number of maneuvers. On the other hand if the clearance is small, the collision should be avoided by using smaller Δq.

Everything is now in place to be integrated in the software development kit we are currently developing. The main critical issue was to devise a well-grounded steering method. The main components of the software integration are:

- *Link the defined steering method within a sampling planning scheme.* The sampling based motion planning methods require three main operators: a steering method, a collision checker and a path optimizer. They build a roadmap whose nodes are collision-free configurations. The edges between two nodes modeling the existence of a collision-free path are computed with the steering method. The geometric considered bodies are made of the box attached to the robot holding it (Fig. 7). Holding configurations of the arms are computed by simple inverse kinematics. Once a first collision-free path is found, it can be optimized to remove detours introduced by the sampling schemes. Here we apply the same type of optimizer as for mobile robots (i.e. based on classical dichotomy procedures).
- *Integrate a walking pattern generator allowing the robot to follow the planned paths.* The pivoting manipulation motion for upper body can be implemented using impedance control to maintain the holding force [2]. The walking pattern generator has already been integrated in another application of humanoid robot combining simultaneous walking and object carrying manipulation behaviors while preserving dynamic balance [17].

VII. Conclusions

In this paper we addressed a motion planning algorithm for pivoting manipulation of a large object by a humanoid robot.

First, we have shown that the pivoting manipulation has the property of small-time controllability that is important to be integrated in sampling-based motion planning schemes.

Next, we have introduced a steering method for pivoting and also shown its small-time controllability. To enable the steering method to connect any configurations, another steering method of Reeds and Shepp wheeled robot is employed. This Reeds and Shepp path is suitable to take account of humanoid robot whole-body manipulation since the robot can always face the moving direction. Then we have presented an algorithm to transform the planned path into a sequence of pivoting operations. The effectiveness of the proposed algorithm has been verified through simulations using HRP-2 model.

As the general policy of planning of pivoting manipulation has been established in this paper, we now envision the planning in the environment populated by obstacles and experimental verification of the proposed method using a humanoid robot platform.

Finally manipulating bulky objects may require changing the holding strategy according to the context. For instance there may be no collision-free path for the robot facing a given side of the object, while there is one for the robot facing another side. This requires holding and re-holding planning. This problem, known as the manipulation *task* planning problem ([18], Chapter 11), can be addressed by processing multiple roadmaps, one for each possible grasping posture. Sampling methods for manipulation task planning apply as well as sampling methods for path planning [11], even if they are more complicated with respect to software integration.

Appendix: computation of Lie Bracket for pivoting motion

Let us consider the vector field L associated to left turn.

After left turn of ε around A (counterclockwise) in Fig. 3, the final configuration is:

$$
\begin{aligned}
x' &= x + l\cos\theta(\cos\varepsilon - 1) - l\sin\theta\sin\varepsilon, \\
y' &= y + l\sin\theta(\cos\varepsilon - 1) - l\cos\theta\sin\varepsilon, \\
\theta' &= \theta + \varepsilon.
\end{aligned}
$$

Then its vector field is given as:

$$
L = \begin{pmatrix} \lim_{\varepsilon\to 0} \dfrac{x' - x}{\varepsilon} \\ \lim_{\varepsilon\to 0} \dfrac{y' - y}{\varepsilon} \\ \lim_{\varepsilon\to 0} \dfrac{\theta' - \theta}{\varepsilon} \end{pmatrix}
$$

where

$$
\begin{aligned}
\frac{x' - x}{\varepsilon} &= \frac{l(\cos(\theta+\varepsilon) - \cos\theta)}{\varepsilon}, \\
\frac{y' - y}{\varepsilon} &= \frac{l(\sin(\theta+\varepsilon) - \sin\theta)}{\varepsilon}, \\
\frac{\theta' - \theta}{\varepsilon} &= \frac{(\theta+\varepsilon) - \theta}{\varepsilon}.
\end{aligned}
$$

When $\varepsilon \to 0$, $\frac{x'-x}{\varepsilon}$ tends to $-\sin\theta$. In the same way, if $\varepsilon \to 0$, $\frac{y'-y}{\varepsilon}$ and $\frac{\theta'-\theta}{\varepsilon}$ tends $\cos\theta$ and 1 respectively.

Then the vector field of at configuration $q(x, y, \theta)$ is

$$
L = \begin{pmatrix} -l\sin\theta \\ l\cos\theta \\ 1 \end{pmatrix}
$$

With the same computation, we show that the vector field for right turn R (counterclockwise):

$$
R = \begin{pmatrix} l\sin\theta \\ -l\cos\theta \\ 1 \end{pmatrix}
$$

Then the pivoting control system is:

$$
\dot{q} = Lu_1 + Ru_2
$$

where u_1, u_2 are the input angle for the motion. Notice that u_1 and u_2 cannot be applied simultaneously.

Now we aim to prove the property that the pivoting system is small-time controllable. It suffices to apply Lie Algebra Rank Condition by computing the Lie Bracket $[L, R]$. Let us recall that the k th component $[f, g]_k$ of Lie Bracket of vector field f and g is calculated by:

$$
[f, g]_k = \sum_{i=1}^{n} \left(g_i \frac{\partial f_k}{\partial q_i} - f_i \frac{\partial g_k}{\partial q_i} \right)
$$

where q_i is the i th component of the configuration [7].

By applying this formula to our pivoting system, we obtain

$$
[L, R] = \begin{pmatrix} -2l\cos\theta \\ -2l\sin\theta \\ 0 \end{pmatrix}.
$$

As the family $\{L, R, [L, R]\}$ is linearly independent, then the system is small-time controllable. □

REFERENCES

[1] Y. Aiyama, M. Inaba, and H. Inoue: "Pivoting: A new method of graspless manipulation of object by robot fingers," *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 136 - 143, 1993.

[2] E. Yoshida, P. Blazevic, V. Hugel, K. Yokoi and K. Harada: "Pivoting a large object: whole-body manipulation by a humanoid robot," *J. of Applied Bionics and Biomechanics*, **3**-3, 227-235, 2006.

[3] M. Mason: "Mechanics and Planning of Manipulator Pushing Operation," Int. J. Robotics Research, **5**-3, 53-71, 1986.

[4] K. Lynch: "The Mechanics of Fine Manipulation by Pushing," Proc. IEEE Int. Conf. on Robotics and Automation, 2269-2276, 1992.

[5] A. Bicchi, Y. Chitour, and A. Marigo: "Reachability and steering of rolling polyhedra: a case study in discrete nonholonomy," IEEE Trans. on Automatic Control, **49**-5, 710-726, 2004.

[6] J.-P. Laumond (editor), "Robot Motion Planning and Control," Lectures Notes in Control and Information Sciences 229, Springer, 1998.

[7] H. Sussmann, "Lie brackets, real analyticity and geometric control," Differential Geometric Control Theory (R. Brockett, R. Millman and H. Sussmann, eds.), Vol. 27 of Progress in Mathematics, 1-116, Michigan Technological University, Birkhauser, 1982.

[8] S. LaValle, "Planning Algorithm," Cambridge University Press, 2006.

[9] H. Choset, K. Lynch, S. Hutchinson, G. Kantor, W. Burgard, L. Kavraki and S. Thrun, "Principles of Robot Motion: Theory, Algorithms, and Implementation," MIT Press, 2006.

[10] S. Sekhavat and J-P. Laumond, "Topological property for collision-free nonholonomic motion planning: the case of sinusoidal inputs for chained form systems," *IEEE Trans. on Robotics and Automation*, **14**-5, 671-680, 1998.

[11] T. Simeon, J-P. Laumond, J. Cortes, A. Sahbani, "Manipulation planning with probabilistic roadmaps," *Int. J. Robotics Research*, **23**, 7-8, 729-746, 2004.

[12] J. A. Reeds and R. A. Shepp, "Optimal paths for a car that goes both forwards and backwards," *Pacific Journal of Mathematics*, **145**-2, 367-393, 1990.

[13] P. Soueres and J.-P. Laumond, "Shortest paths synthesis for a car-like robot," *IEEE Trans. on Automatic Control*, **41-5**, 672-688, 1996.

[14] J-P. Laumond, "Kineo CAM: a success story of motion planning algorithms," *IEEE Robotics & Automation Magazine*, **13**-2, 90-93, 2006.

[15] S. LaValle and J. Kuffner, "Rapidly-Exploring Random Trees: Progress and Prospects," *In Algorithmic and Computational Robotics: New Directions*, 293-308, A K Peters, 2001.

[16] L. Kavraki, P. Svestka, J.-C. Latombe, M. Overmars, "Probabilistic Roadmaps for Path Planning in High-Dimensional Configuration Spaces," *IEEE Trans. on Robotics and Automation*, **12**-4, 566-580, 1996.

[17] E. Yoshida, C. Esteves, T. Sakagughi, J-P. Laumond and K. Yokoi, "Smooth Collision Avoidance: Practical Issues in Dynamic Humanoid Motion," *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 827-832, 2006.

[18] J-C. Latombe, "Robot Motion Planning," Kluwer Academic Publishers, 1991.