

Optimized Trajectory Generation based on Model Predictive Control for Turning Over Pancakes

Toshiaki Tsuji* Senior Member, Kyo Kutsuzawa* Student Member
Sho Sakaino* Senior Member

(Manuscript received Nov. 11, 2016, revised May 22, 2017)

This study investigates kinodynamic object manipulation by a robot using a tool. Based on the conditions for maintaining contact between a held spatula and a manipulated object, a variety of movements satisfying such conditions are planned. Model predictive control is introduced to plan an optimal trajectory. Simulation results show that the proposed method plans a variety of turning over motions with different setups of cost functions. Experimental results demonstrate that the trajectory optimization method accomplishes turning over motion, which is a typical example of motion with kinodynamic constraints.

Keywords: haptics, manipulation, kinodynamics, trajectory planning

1. Introduction

A robot is more advantageous than ordinary machines in respect of capability for executing a diverse range of operations. The types of operations that can be performed by a robot depend on the shape of the end effectors and as such, having the ability to change the shapes of the end effectors to a diverse range of shapes is desired. Therefore, studies for making robots handle tools have already been conducted^{(1)–(4)}. Studies on the theory for robots to learn the characteristics of tools are being conducted in the field of cognitive robotics⁽¹⁾. Nabeshima and Kuniyoshi proposed a method for achieving sustainable sensory-motor coordination for robotic tool use⁽²⁾. Beetz *et al.* prepared pancakes using two units of robots to demonstrate their performance⁽³⁾. One of the solutions is to handle numerous tools in a unified manner by analyzing physical interactions of tools. Matsuzaki *et al.* focused on the contact status of a tool with an object when a tool was used, and classified the position, attitude and freedom of such a tool used into four categories⁽⁴⁾.

In order to realize a robot utilizing tools in real environment, some issues have to be solved. Among them, motion planning seems one of the most crucial issues for tool-use robots. Motion planning is well studied in the area of the whole robotics, instead of tool-use robots. Rapidly exploring randomized trees (RRTs)⁽⁵⁾ and probabilistic roadmaps (PRMs)⁽⁶⁾ are cited as the most popular methods for motion planning. Based on these studies, some researches have been conducted on the manipulation considering kinodynamic constraints, that is, the constraints owing to the physical laws the robots are subject to. Kinodynamic planning methods have been applied to humanoid robots⁽⁷⁾, underactuated robots⁽⁸⁾, and quadrotor helicopter⁽⁹⁾. A motion planning

with acceleration limits have been studied as well⁽¹⁰⁾. Model predictive control (MPC) is also a well developed theory, which has an ability to deal with optimal control problems in complicated systems. Therefore, there are examples on motion planning of constrained dynamic systems⁽¹¹⁾ and biped robots⁽¹²⁾⁽¹³⁾.

Although the theories on kinodynamic planning are well developed, there are limited number of studies extending the method to a high performance of a robot in real environments. Andreasson *et al.* have developed a path smoothing approach based on a lattice-based motion planner for non-holonomic industrial vehicles, highly accurate in real environment⁽¹⁴⁾. An MPC based omnidirectional robot has shown good performance of tilt recovery⁽¹⁵⁾. The authors' group has derived the conditions for manipulating an object on a spatula and realized turning over movement of a pancake by a robot holding a spatula⁽¹⁶⁾. Object manipulation of a tool-use robot is selected as the typical example of application with kinodynamic constraints for keeping contact between a tool and an object. Tool-use robots also have a limitation that no sensors should be mounted on the tool. Turning over pancakes requires rapid motion considering the limitation in dynamics. Additionally, it should be conducted in less than a second and quasi-static motions are guaranteed to fail. Therefore, kinodynamic object manipulation using spatula is a good candidate to verify practical performance. Although the study in (16) shows that turning over trajectory is automatically generated, a methodology to optimize the trajectory for some specific objectives remained unclear. Therefore, this study introduces Model Predictive Control (MPC) to the turning over motion. A method to generate a trajectory optimizing for some objectives under kinodynamic constraints is shown. Additionally, simulation and experimental results show the validity of the method.

* Saitama University
255, Shimo-okubo, Sakura-ku, Saitama 338-8570, Japan

2. Motion Planning Scheme

The motion planning in this study is based on MPC. Although it has been mentioned that MPC needs larger calculation amount with high dimensional systems⁽¹⁷⁾, this study avoids the issue by introducing a simplified model with reduced dimension. Firstly, simplified models for online motion planning based on MPC are introduced in 2.1. Secondly, the scheme based on the model is described in 2.2.

2.1 Models of Object Dynamics and Contact The predictive control in this study focuses on the motion of the object. The dynamics model of the object and the contact model between the model and the spatula is derived. While this is intended for robots in a three-dimensional space, the fundamental principles are described with movements within a plane, since the movements of turning over and sliding objects can both be described as movements on a two-dimensional plane. The object motion should satisfy the constraints given from the two models. The dynamics model of the spatula and the robot is not considered for simplicity of the model. Figure 1 shows the models. Firstly, the object is modeled as a rigid body and its dynamics is described as follows:

$$\mathbf{F} = m\mathbf{a} - m\mathbf{g} \dots \dots \dots (1)$$

where $\mathbf{m} = \text{diag}(m, m, I_\theta)$, $\mathbf{a} = [a_y, a_z, a_\theta]^T$, and $\mathbf{g} = [0, g_z, 0]^T$. m and I_θ are the mass of the object and moment of inertia (MOI) in θ rotation, respectively. θ is an inclination angle of the object with respect to the horizontal plane. a is acceleration, while the subscripts y and z denote horizontal and vertical axes. g_z is gravity acceleration.

2.2 Model Predictive Control Although nonlinear MPC schemes are accessible to fast systems requiring short computation times, linear MPC schemes still allow shorter computation times. Under the assumption that the contact between the object and the spatula is maintained, the acceleration of the object will be equal to that of the robot. Then, its dynamics are described as follows:

$$\hat{\mathbf{x}}(k+1) = \mathbf{A}\hat{\mathbf{x}}(k) + \mathbf{B}\mathbf{u}(k) \dots \dots \dots (2)$$

$$\hat{\mathbf{x}}(k) = [y(k), z(k), \theta(k), v_y(k), v_z(k), v_\theta(k)]^T \dots \dots \dots (3)$$

$$\hat{\mathbf{u}}_k = [a_y(k), a_z(k), a_\theta(k)]^T \dots \dots \dots (4)$$

$$\mathbf{A} = \begin{bmatrix} 1 & 0 & 0 & \Delta T & 0 & 0 \\ 0 & 1 & 0 & 0 & \Delta T & 0 \\ 0 & 0 & 1 & 0 & 0 & \Delta T \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

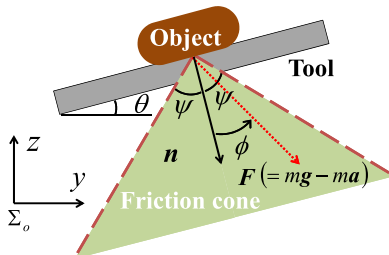


Fig. 1. Force acting on the object placed on a tool

$$\mathbf{B} = \begin{bmatrix} \Delta T^2/2 & 0 & 0 \\ 0 & \Delta T^2/2 & 0 \\ 0 & 0 & \Delta T^2/2 \\ \Delta T & 0 & 0 \\ 0 & \Delta T & 0 \\ 0 & 0 & \Delta T \end{bmatrix}$$

Here, $\hat{\mathbf{x}}_k$ is the k th sample of the state of the object. ΔT is the sampling time.

As discussed later, the condition to maintain the contact is based on the acceleration of the tool. Since MPC is an effective method to optimize the trajectory under limitation of the input, this study introduces MPC. The proposed MPC method computes a trajectory of the object which minimizes a cost function related with each movement. For example, the cost function of the turning over pancake is as follows:

$$\min \left((\theta(k_g) - \pi)^2 + \alpha (y(k_g) - y^{cmd})^2 + \beta (v_y(k_g)^2 + v_z(k_g)^2) \right). \dots \dots \dots (5)$$

Here, k_g denotes the sample that the object (i.e. pancake) touches the ground after the turning over movement. α and β denote the weighting factors on landing position error and landing velocity, respectively. If α is larger, the weight of landing position in the cost function becomes larger. As a result, the optimized trajectory has a smaller landing position error. On the other hand, the weight of landing velocity becomes larger as β is larger, which results in a landing with smaller velocity. The cost calculated by (5) depends on the object angle θ , the horizontal position y , horizontal velocity v_y , and vertical velocity v_z , at the k_g th sample. θ is a variable to determine the success or failure of the turning over motion. y shows where the object drops and its proper position is often required in practice. v_y , and v_z are the velocity in each axis and they should be small in most cases to avoid smashing owing to the impact. Therefore, these four variables are included in the cost function.

The procedures of the MPC are as follows:

- (1) determine the candidate of the input over the prediction horizon,
- (2) calculate the predicted state, and
- (3) evaluate the cost function.

After repeating procedures (1)–(3) with different possible input, the input with the lowest cost is selected.

3. Conditions for MPC

MPC described in the previous section finds an optimal trajectory after repeating the calculation with different inputs. It is quite common that the trajectory is derived by an analytical calculation. However, this study applies numerical analysis based on Eqs. (2)–(4), since numerical analysis shows higher flexibility to variable input in prediction horizon. On the other hand, the number of candidate inputs should be limited in such a method to reduce the calculation amount. The conditions of the inputs, which maintain the contact between the object and the spatula can be calculated from the contact model. Therefore, the following section describes the way to limit the number of candidate inputs from the contact model.

3.1 Model of Contact between Object and Tool

Supposing the object on the tool starts to slide on the tool at

$\theta \geq \psi$, the maximum static frictional force F_{imax} is expressed by the following formula, using the static friction coefficient μ :

$$F_{imax} = \mu N = \mu mg \cos \psi. \dots \dots \dots (6)$$

In the “turning over pancakes”, the tool will necessarily be in an inclination that exceeds the friction angle ψ . In such an instance, some kind of force must be applied to the object to ensure the contact.

The sliding condition of the object is considered by using a friction cone depicted in Fig. 1. If we let the vector perpendicular to the surface of the tool to be \mathbf{n} and define the angle created by \mathbf{F} and \mathbf{n} to be ϕ , then the condition for each contact state can be given in the following manner, by using ϕ :

$$0 \leq \phi < \psi. \dots \dots \dots (7)$$

$$\psi \leq \phi < \frac{1}{2}\pi \dots \dots \dots (8)$$

$$\frac{1}{2}\pi \leq \phi \dots \dots \dots (9)$$

where (7), (8), and (9) are the conditions for the object in contact without sliding, the object sliding, and the object released from the spatula, respectively.

Figure 2 conceptually depicts the conditions of each contact state (7), (8), and (9). The “turning over pancakes” can be achieved by planning a trajectory satisfying (7) until the inclination angle θ is over $\frac{1}{2}\pi$. The object slides over the tool with the maximum static frictional force being exceeded when (7) are not satisfied. Such a trajectory should be planned in some specific cases. For example, in case the pancake needs to slide on the spatula to lay it on a plate, the trajectory should satisfy (8). The relationship between the tool and the object shall hereinafter be referred to as the sliding angle ϕ . These conditions apply to both statics and dynamics. The difference of these two is existence of dynamic terms such as inertia force. In other words, in case the static gravity force does not satisfy (7), the contact of the object can be still maintained by adjusting the inertia force. Such a method is described in

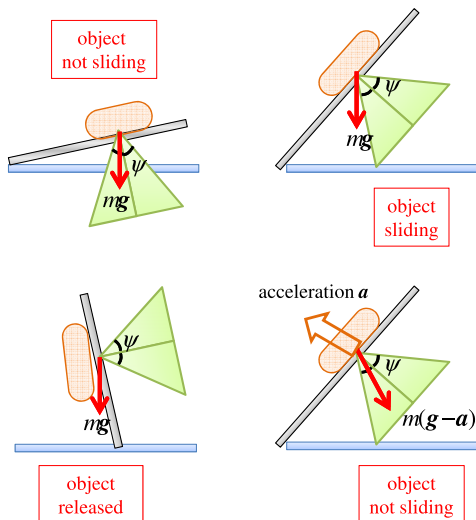


Fig. 2. Relationship between inclination of tool and sliding of object

Section 3.2.

Equation (4) is given under the assumption that the object maintains the contact with spatula. Therefore, the formula should be replaced if (7) is not satisfied. When (8) is satisfied, the object slips on the spatula. Then, (4) is rewritten as:

$$\hat{\mathbf{u}}_k = [a_y(k) \sin \theta, a_z(k) \cos \theta, a_\theta(k)]^T \dots \dots \dots (10)$$

Otherwise, (9) should be satisfied. θ is the angle of the object, while it is often difficult to measure the angle of the object in the real environment. Therefore, input $\hat{\mathbf{u}}_k$ is determined based on the tool angle instead of the object angle. Since (9) is a condition that the object is released, the following input should be given in such a case.

$$\hat{\mathbf{u}}_k = [0, g, 0]^T \dots \dots \dots (11)$$

3.2 Acceleration Limit based on Inertial Force

Let us consider a situation where the tool is accelerating, as shown in Fig. 3, in order to derive the force vector \mathbf{F} that satisfies (7). Let us express the resultant force of the gravitational force acting on the object $m\mathbf{g}^{tr}$ and the inertial force $-m\mathbf{a}^{tr}$ as $m\mathbf{i}$.

$$m\mathbf{g}^{tr} + (-m\mathbf{a}^{tr}) = m\mathbf{i} \dots \dots \dots (12)$$

Here, \mathbf{i} is a contact acceleration. $\mathbf{g}^{tr} = [0, g_z]^T$ and $\mathbf{a}^{tr} = [a_y, a_z]^T$ are the gravity vector and the acceleration vector of the object considering only translational components. The rotational component is neglected here, because it hardly affects the contact state as long as the height of the object is small. Since $m\mathbf{i}$ is equivalent to external force, it can be said that the vector physically expresses the external force in units of acceleration. The vectors parallel and perpendicular to the tool surface are defined as \mathbf{s} and \mathbf{n} , respectively.

The sliding angle ϕ is the angle created by \mathbf{n} and \mathbf{i} . Incidentally, when considering the sliding angle $\phi = 0$, the following equation is derived:

$$\mathbf{i}_{\phi=0} = [i_y, i_z]^T = [I \sin \theta(k), -I \cos \theta(k)]^T \dots \dots \dots (13)$$

where I is a scalar value of contact acceleration \mathbf{i} . By utilizing (13), vector \mathbf{i} can be generally expressed using the following equation:

$$\mathbf{i} = [i_y, i_z]^T = [I \sin(\theta(k) + \phi), -I \cos(\theta(k) + \phi)]^T \dots \dots \dots (14)$$

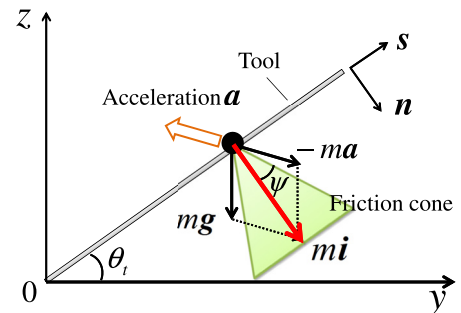


Fig. 3. Acting force during acceleration and friction cone

By substituting (14) to (12), the ideal acceleration satisfying desired I and ϕ is derived as follows:

$$Veca^{tr} = [a_y, a_z]^T = [-I \sin(\theta(k) + \phi), g_z + I \cos(\theta(k) + \phi)]^T \quad (15)$$

If the acceleration of (15) can be achieved so that ϕ satisfies (7) as described above, “turning over pancakes” for any inclination $\theta(k)$ of the tool becomes possible, without the object sliding on the tool.

Such possible inputs are calculated from different patterns of I and ϕ within the condition of the desired motion. The MPC scheme calculates the trajectory over prediction horizon from the time series of candidate inputs. Since there are numberless patterns of possible inputs, some typical patterns are selected from the condition of the parameters. For example, the following parameters were used to generate the inputs over prediction horizon when the contact between a spatula and a pancake has to be maintained.

$$\phi = -0.8\psi, 0, 0.8\psi \quad (16)$$

$$I = -1.5g_z, -g_z, -0.5g_z \quad (17)$$

On the other hand, the input to slide an object was given by the following equations.

$$\phi = \psi + \left(\frac{\pi}{2} - \psi\right) / 2 \quad (18)$$

$$I = -1.5g_z, -g_z, -0.5g_z \quad (19)$$

At the end of the turning motion, the pancake should be separated from the spatula. Therefore, the inputs to release the object should also be considered. Such inputs are derived as follows:

$$\phi = 0 \quad (20)$$

$$I = 0.5g_z \quad (21)$$

Figure 4 shows the input patterns. Three horizontal dotted lines denote (16), (18), and (20), while perpendicular and slant dotted lines denote (17), (19), and (21). Filled circles on the crosspoints of these lines are the input patterns.

In case of halting a robot, the acceleration reducing velocity should be given as follows:

$$\phi = \tan^{-1}(v_y/v_x) \quad (22)$$

$$I = 0.5g_z \quad (23)$$

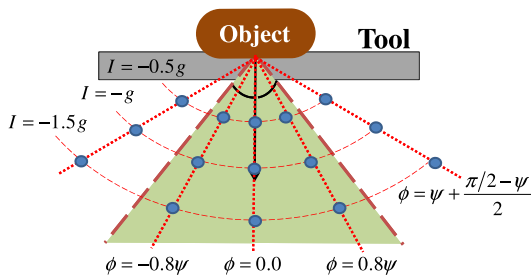


Fig. 4. Candidate of inputs

3.3 Mechanical Constraints The previous subsection described the way to calculate the input over the prediction horizon with acceleration satisfying the contact condition. Other to the contact condition, it is necessary to take note of the mechanical constraints of the robot. The following three aspects are considered as mechanical constraints of the robot that should be noted.

Constraint (a): movable range of robot.

Constraint (b): performance limits of respective articulating motors (maximum velocities, etc.)

Constraint (c): singular configuration of robot.

As described in 2.2, the first procedure in this method is to determine a candidate of inputs. The input is determined based on the calculation in Section 2.1. Then, the cost function is evaluated from the state calculation by MPC. In case one of the states exceeds the abovementioned constraints, the MPC calculation is aborted at the middle of the prediction horizon and the cost function of the horizon is set to be maximum. Then, the trajectories beyond the kinematic constraint is eliminated.

The abovementioned way is effective for eliminating inputs resulting in excess of the mechanical constraints, while there is an issue that calculation time is wasted for them. If there are mechanical constraints directly related with inputs, it is also possible to reduce the calculation time by selecting the inputs satisfying part of the mechanical constraints. For example in the case of the “turning over pancakes”, the inclination velocity of the tool v_θ is given first to make the optimization simple. Since the maximum velocity and the inertial of the robot wrist is both quite small, it is possible to rotate the wrist joint, strongly related with inclination of the object, with maximum velocity during all the turning over motion. Thus $\theta(k)$ is given as a function of time with the following equation:

$$\theta(k) = v_\theta^{\max} t = v_\theta^{\max} k \Delta T \quad (24)$$

where v_θ^{\max} [rad/s] is the maximum of v_θ .

4. Verification in Experiment and Simulation

This section describes the results of experiment and simulation. First, simulation results show how the optimized trajectory varies depending on the parameters in the cost function. It is difficult to measure the position of the pancake and the reaction force on the spatula in experiment. Therefore, the landing position and the angle of the reaction force were also evaluated in the simulation. Second, experimental results verify whether kinodynamic motion was generated in real environment.

4.1 Setup of the Robot The robot used in this study is a 6-DOF manipulator “Motoman-MH3F”, supplied by Yaskawa Electric. The weight of the main unit of the robot is 27 [kg]. A six-axial force sensor (WDF-6M200-3 supplied by Wacoh-Tech Inc.) is mounted on the tip of the arm that enabled measurements of the tri-axial forces transmitted on the tip of the arm F_x , F_y , and F_z , as well as the tri-axial moments M_x , M_y , and M_z . A gripping mechanism was also installed via a force sensor to configure a structure for holding a tool. PD controller with disturbance observer⁽¹⁸⁾ with 1 [ms] control period is implemented. For simplicity, it was assumed that the response of the spatula follows the

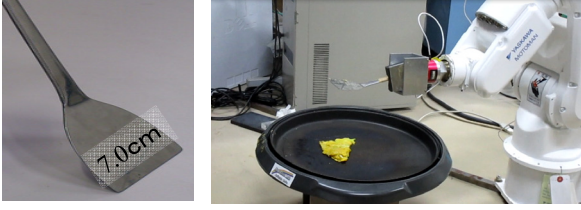


Fig. 5. Photo of the spatula and the robot

command accurately in the simulation. A PD controller with disturbance observer⁽¹⁸⁾ was implemented to the experimental system. The proportional position gain K_{pp} , proportional attitude gain K_{pr} , derivative position gain K_{vp} , and derivative attitude gain K_{vr} were 150.0, 400.0, 10.0, and 40.0, respectively. G_{DOB} , the cutoff frequency of disturbance observer was 2.0 [rad/sec] and force gain K_f was 0.15, while K_f was set to 0 during the dynamic motion in order to accomplish accurate acceleration by position control.

4.2 Evaluation in Simulation First, the simulation results of turning over motion was evaluated. Dynamics of the robot was simulated in Open Dynamics Engine (ODE) with 0.1 msec sampling time. The trajectory planning assumed that the height of the ground was 0.2 m. Therefore, the position and velocity when the height of the pancake was 0.2 m was evaluated. Figs. 6 and 7 show the result of optimized trajectory of turning over pancakes motion generated by the proposed method and compare the result with different parameters. As the initial position in y axis was 0, y^{cmd} was also set to 0. Position of the center of mass (COM) was plotted. The stick diagram of the spatula movement is overlapped to the trajectory. The both figures show that the curve of the pancake accorded with that of the spatula in the beginning, while the acceleration was determined based on the contact condition. After the trajectory generation based on MPC, the speed of the spatula was reduced so that the acceleration satisfies (9). Then, the pancake was released. Note that the acceleration satisfies (9) in most cases of a simple speed reduction because the spatula is moving downward in the end of the turning over motion. β was fixed to 0.5 in the experiments on Fig. 6. The pancake angle at landing $\theta(k_g)$, the landing position $y(k_g)$, and landing velocity $|v(k_g)|$ with different parameters are compared in Fig. 8. Here, $v(k_g) = (v_y(k_g), v_z(k_g))$. Since α is the weighting factor of the error in landing position, the landing position error reduced as α increased. Figure 7 shows trajectories with different β . α was fixed to 20 then. β is the weighting factor of landing velocity. Therefore, landing velocity reduced as β increased. The turning over motion takes longer time as β is smaller. Since the rate of the landing angle of the pancake in the cost function became larger, a trajectory with longer time (in other words, larger spatula inclination) was selected.

Figure 9 shows the angle ϕ during turning over motion. The blue shaded area shows the range of friction cone. The MPC generates trajectory so that the angle ϕ is in the friction cone. Since ϕ remained in the friction cone, turning over motion was successfully accomplished. In between 25.1 to 25.2 sec, the response angle varied a lot. As shown in Fig. 10, the response acceleration generated error just after the sudden variation of command acceleration. This result shows

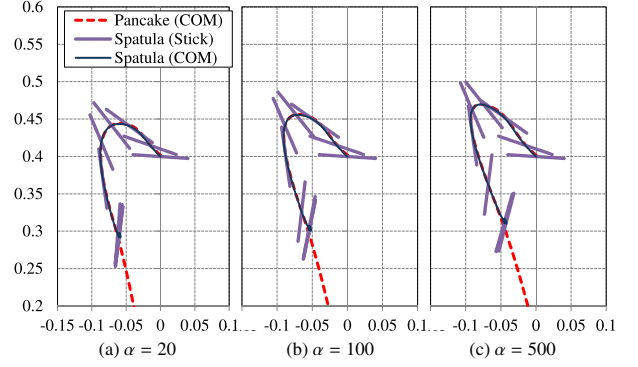
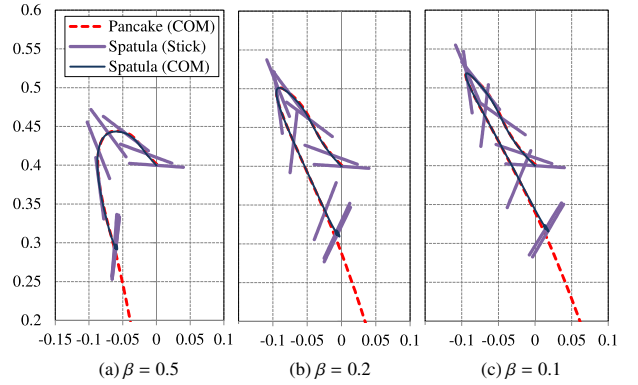
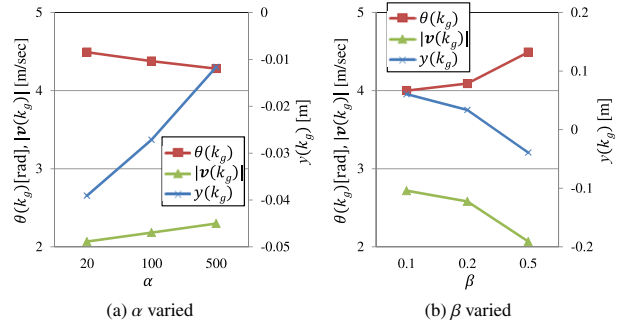
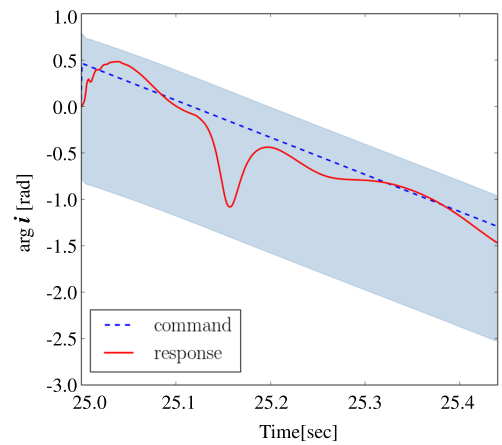

 Fig. 6. Trajectory with different α

 Fig. 7. Trajectory with different β


Fig. 8. Variables related to cost function


 Fig. 9. Angle ϕ during turning over motion

that the error occurs in a frequency range larger than the control bandwidth and trajectory generation with low bandwidth components is desirable.

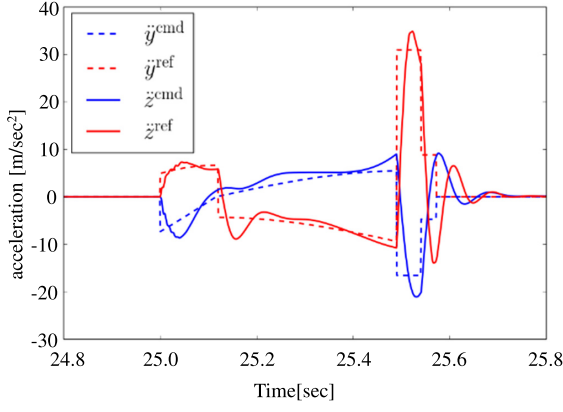


Fig. 10. Acceleration response during turning over motion

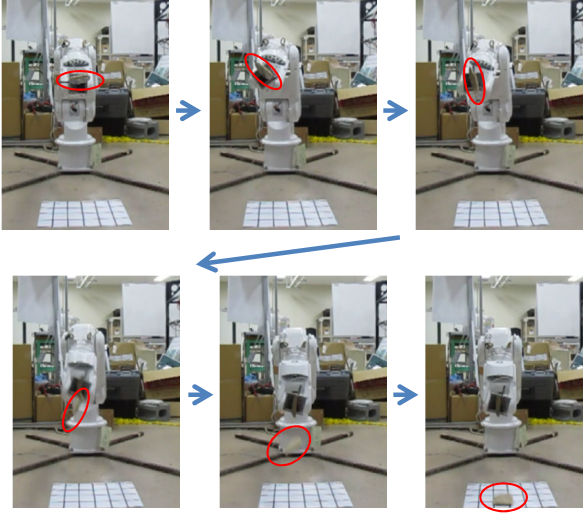


Fig. 11. Snapshots of turning over motion

4.3 Experimental Results The turning over motion was executed in the abovementioned experimental setup. 10 times of turning over motion is executed with the trajectory optimized with $\alpha = 20$ and $\beta = 0.5$. 1 out of 10 results ended up in failure. The pancake slipped away and it was not turned over then. The other 9 results were successful with a pancake turned over. The average error was 2.9 [cm] with 4.4 [cm] standard deviation. Although it is difficult to define a target value of the landing accuracy, the result at least show that the high repeatability of turning over motion quantitatively.

Snapshots of the turning over motion is shown in Fig. 11. It shows that the pancake was in contact with the spatula until 0.5 sec after starting turning over motion, and the timing is almost the same to the simulation results. The position responses and velocity responses are shown Figs. 12 and 13. Although some errors exist in position dimension, the turning over motion was accomplished as long as the acceleration is in a certain range.

5. Conclusion

Turning over a pancake is a typical example of motion with kinodynamic constraints. This study proposed an MPC scheme for trajectory planning of a robot turning over pancakes. Simulation results demonstrated that the optimized trajectory accomplishes turning over motion. The MPC

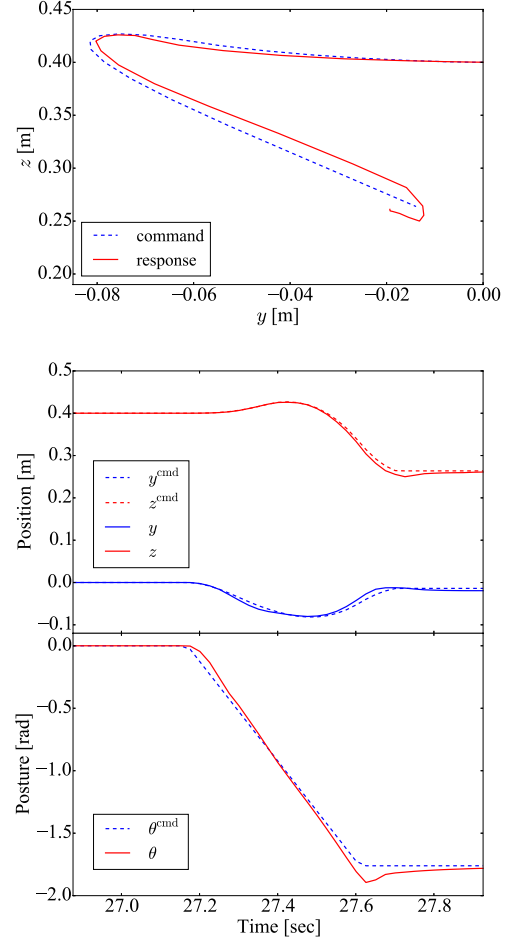


Fig. 12. Position response of turning over motion

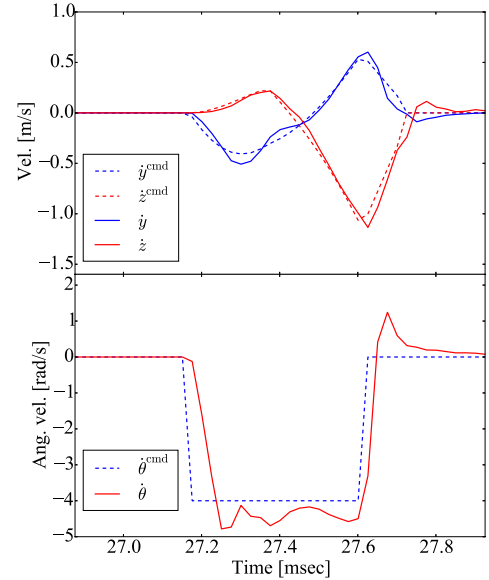


Fig. 13. Velocity response of turning over motion

scheme has an ability to adjust the trajectory for cost functions with different purposes.

Acknowledgment

This study was supported in part by the Grant-in-Aid for Scientific Research (B), Number 16H04311.

References

- (1) A. Stoytchev: "Behavior-grounded representation of tool affordances", in Proc. IEEE Int. Conf. on Robotics and Automation, pp.3071–3076 (2005)
- (2) C. Nabeshima and Y. Kuniyoshi: "A Method for Sustaining Consistent Sensory-Motor Coordination under Body Property Changes Including Tool Grasp/Release", *Advanced Robotics*, Vol.24, No.5–6, pp.687–717 (2010)
- (3) M. Beetz, et al.: "Robotic Roommates Making Pancakes", in Proc. IEEE 11th IEEE-RAS Int. Conf. on Humanoid Robots, pp.529–536 (2011)
- (4) R. Matsuzaki, M. Kamibayashi, S. Sakaino, and T. Tsuji: "Classification of a Hybrid Control System for Robotic Tool Use", in Proc. IEEE Int. Conf. on Mechatronics (ICM), IF-002712, pp.712–717 (2013)
- (5) J.J. Kuffner and S.M. LaValle: "RRT-connect: An efficient approach to single-query path planning", in Proc. IEEE Int. Conf. on Robotics and Automation, pp.995–1001 (2000)
- (6) L.E. Kavraki, P. Svestka, J.C. Latombe, and M.H. Overmars: "Probabilistic roadmaps for path planning in high-dimensional configuration spaces", *IEEE Trans. on Robotics and Automation*, Vol.12, No.4, pp.566–580 (1996)
- (7) J.J. Kuffner, S. Kagami, K. Nishiwaki, M. Inaba, and H. Inoue: "Dynamically-stable motion planning for humanoid robots", *Autonomous Robots*, Vol.12, No.1, pp.105–118 (2002)
- (8) G. Gortekin, A. Perez, R. Platt, and G. Konidaris: "Optimal Sampling-Based Planning for Linear-Quadratic Kinodynamic Systems", in Proc. IEEE Int. Conf. on Robotics and Automation (ICRA2013), pp.2429–2436 (2013)
- (9) D. Webb and J. van den Berg: "Kinodynamic RRT*: Asymptotically optimal motion planning for robots with linear dynamics", in Proc. IEEE Int. Conf. on Robotics and Automation (ICRA2013), pp.5054–5061 (2013)
- (10) T. Kunz and M. Stilman: "Probabilistically complete kinodynamic planning for robot manipulators with acceleration limits", in Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS2014), pp.3713–3719 (2014)
- (11) D. Nguyen-Tuong and J. Peters: "Model learning for robot control: a survey", *Cognitive processing*, Vol.12, No.4, pp.319–340 (2011)
- (12) P.B. Wieber, "Trajectory free linear model predictive control for stable walking in the presence of strong perturbations", in Proc. 2006 6th IEEE-RAS Int. Conf. on Humanoid Robots, pp.137–142 (2006)
- (13) H. Diedam, D. Dimitrov, P.B. Wieber, K. Mombaur, and M. Diehl: "Online walking gait generation with adaptive foot positioning through linear model predictive control", in Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, pp.1121–1126 (2008)
- (14) H. Andreasson, J. Saarinen, M. Cirillo, T. Stoyanov, and A.J. Lilienthal: "Fast, continuous state path smoothing to improve navigation accuracy", in Proc. IEEE Int. Conf. on Robotics and Automation, pp.662–669 (2015)
- (15) J. Lafaye, C. Collette, and P.B. Wieber: "Model predictive control for tilt recovery of an omnidirectional wheeled humanoid robot", in Proc. IEEE Int. Conf. on Robotics and Automation, pp.5134–5139 (2015)
- (16) T. Tsuji, J. Okuma, and S. Sakaino: "Dynamic Object Manipulation Considering Contact Condition of Robot with Tool", *IEEE Trans. on Industrial Electronics*, Vol.63, No.3, pp.1972–1980 (2016)
- (17) S. Karaman and E. Frazzoli: "Sampling-based algorithms for optimal motion planning", *International Journal of Robotics Research*, Vol.30, No.7, pp.846–894 (2011)
- (18) T. Murakami, F. Yu, and K. Ohnishi: "Torque Sensorless Control in Multidegree-of-Freedom Manipulator", *IEEE Trans. on Industrial Electronics*, Vol.40, No.2, pp.259–265 (1993)

Toshiaki Tsuji

(Senior Member) received the B.E. degree in system design engineering and the M.E. and Ph.D. degrees in integrated design engineering from Keio University, Yokohama, Japan, in 2001, 2003, and 2006, respectively. He was a research associate in Department of Mechanical Engineering, Tokyo University of Science from 2006 to 2007. He is currently an Associate Professor in Department of Electrical and Electronic Systems, Saitama University. His research interests include motion control, haptics and rehabilitation robot. He received the FANUC FA and Robot Foundation Original Paper Award in 2007 and 2008.

Kyo Kutsuzawa

(Student Member) received the B.E. degree in electrical and electronic systems and the M.E. degree in science and engineering from Saitama University, Japan, in 2015 and 2017, respectively. He is currently working on Ph.D. degree at Graduate School of Science and Engineering, Saitama University. His research interests include tool-using robot, haptics, and neural networks.

Sho Sakaino

(Senior Member) received the B.E. degree in system design engineering and the M.E. and Ph.D. degrees in integrated design engineering from Keio University, Yokohama, Japan, in 2006, 2008, and 2011, respectively. He is currently an assistant professor with the Department of Electrical and Electronic Systems, Saitama University, Saitama, Japan. His research interests include mechatronics, motion control, robotics, and haptics. He received IEEE Industry Application Society Distinguished Transaction Paper Award in 2011.