Joint Torque Optimization for Quasi-static Graspless Manipulation

Satoshi Makita and Yusuke Maeda

Abstract— Graspless manipulation is easily interfered by external disturbances because the manipulated object is not completely held by a robot hand and supported by an environment such as a floor. Thus it is important to ensure the manipulation is executed robustly against some disturbances. In our works, we have proposed a rigid-body-based analysis of indeterminate contact forces for quasi-static graspless manipulation, and also joint torque optimization for robotic hands. The joint torques of the robot is determined in consideration of some robustness of manipulation against disturbances, which include changes or estimation errors of friction. In the analysis of contact forces in quasi-statics, we consider a kinematic constraint on static friction to exclude infeasible sets of frictional force, with considering treatment of kinetic friction. We also propose new objective functions for computing optimal joint torques in both static and quasi-static graspless manipulation. Some numerical samples of both applications are shown to verify our proposed methods.

I. Introduction

In graspless manipulation [1] (or nonprehensile manipulation [2]) such as pushing, pivoting and tumbling (Fig. 1), the manipulated object is not only held by a robot hand, but also supported by the environment. These manipulations have some advantages over typical grasping and pick-and-place. The robot has not support all the weight of the object while it is manipulating, and can deal with heavier objects than what the robot can pick up. Thus required joint torque of robots can be decreased in graspless manipulation over conventional grasping, for example, comparing pick-and-place and sliding manipulation in Fig. 1.

We have studied the analysis of contact forces in graspless manipulation for robustness measure and joint torque optimization [3], [4]. In the analysis, a constraint on static friction is considered, which is formulated by Omata et al. for power grasps [5], [6]. The constraint is based on the kinematics of contact points under rigid-body mechanism. We modified the formulation to apply it to graspless manipulation with consideration of kinetic friction caused

S. Makita is with Dept. of Control Engineering, Sasebo National College of Technology, 1-1 Okishincho, Sasebo, Japan `makita@sasebo.ac.jp`

Y. Maeda is with Div. of Systems Research, Faculty of Engineering, Yokohama National University `maeda@ynu.ac.jp`
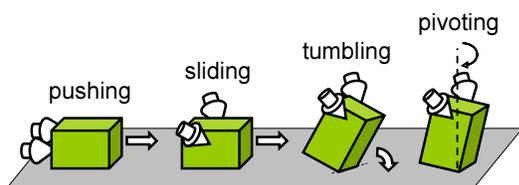
Fig. 1. Examples of graspless manipulation

by the manipulated object's motion at some contact points (or area). The consideration is, however, not sufficient and causes some unreasonable solutions that contact forces in quasi-static manipulation reach infinity. Thus we rectify the consideration of kinetic friction to apply the constraint on static friction to quasi-statics to calculate contact forces in graspless manipulation in this paper. Also we derive algorithms for optimal joint torques in quasi-static graspless manipulation with the new analysis.

Joint torques in static grasping has been studied in many researches. In enveloping grasping [7] or power grasps [8], the contact forces on multiple contact points are indeterminate under rigid-body mechanics, and they can vary depending on the mechanism of the robot hand even with constant joint torques. Thus the robot hand can keep some robustness against external disturbances without changing its joint torques [9]. Yong et al. defined optimal power grasps as grasps that can resist against external disturbances and minimize the maximum joint torque, and computed them [10] Omata et al. proposed two objective functions: maximizing joint torques to resist against as large changes of disturbances as possible; minimizing joint torques that equilibrates contact forces with given disturbances as [10]. These objective functions are based on margins between every friction cone and contact force within it. The margin is, however, directly applied to graspless manipulation because the margin between the friction cone and kinetic friction force is always zero. Thus we propose a new objective function to compute optimal joint torques in graspless manipulation in this paper.

II. Mechanical Model

A. Assumptions

We make the following assumptions for graspless manipulation by multiple robot fingers as [3].

- All the manipulated objects, the robot fingers and the environments are rigid.
- All the contacts including surface contacts can be represented by finite point contacts.
- Coulomb friction among the object, the robot fingers and the environments has occur at every contact points.
- Every friction cone at a contact point is approximated by a polyhedral convex cone [11].
- Each finger is in one of the following control modes: position-control, force-control and hybrid force/position control.
  1) The servo stiffness of position-control can reach infinity. Thus each robot finger in position-control

mode can be regarded as an environment and apply arbitrary force within its friction cone.

2) Each robot finger in force control mode can apply contact force determined by the corresponding joint torques and a Jacobian matrix.

3) Each robot finger in hybrid force/position control can be regarded as a prismatic joint whose fingertip has contact with the object, and apply a commanded normal force actively and an arbitrary tangential force within its friction cone passively.

In addition, we define some notations as follows:

- N : the number of fingers,
- M_i : the number of contact points on the i-th finger,
- $M := \sum_{i=1}^{N} M_i$: the total of contact points,
- L_i : the number of joints of the i-th finger,
- $L := \sum_{i=1}^{N} L_i$: the total of joints,
- P_{ik} : the k-th contact point on the i-th finger,
- P_l : the l-th contact point where $l = \sum_{n=1}^{i-1} L_n + k$,
- $p_l \in \mathbb{R}^3$: the position vector of P_l.

As a matter of formulation, all the contact points on the environment and the palm of the hand are included in those on the 1st finger.

B. Contact forces

When the manipulated object is moving quasi-statically such as in pushing, there are some sliding contact points corresponding to the object's motion. To distinguish contact points in actual sliding, we define the following matrix:

$$D := \mathrm{diag}\,(d_1 I_3, \ldots, d_M I_3) \in \mathbb{R}^{3M \times 3M} \qquad (1)$$

$$d_l := \begin{cases} 1 & \text{when the } l\text{-th contact point is actually sliding,} \\ 0 & \text{otherwise.} \end{cases}$$
$$(2)$$

At a contact point in actual sliding (then $d_l = 1$), kinetic frictional force is applied in the opposite direction to its sliding direction. On the other hand, contact forces including static frictional forces at stationary contact points can be applied within each friction cone. The contact force $f_l \in \mathbb{R}^3$ applied to the object at P_l can be represented as:

$$f_l = C_l k_l, \qquad (3)$$

where

$$C_l := \begin{cases} [c_{l1}] \in \mathbb{R}^{3 \times r_l}, \ r_l = 1 & \text{when } d_l = 1, \\ [c_{l1}, \ldots, c_{lr_l}] \in \mathbb{R}^{3 \times r_l} & \text{when } d_l = 0. \end{cases} \qquad (4)$$

$c_{lm} \in \mathbb{R}^3$ is a unit edge vector of a polyhedral convex cone that approximates the friction cone at P_l; r_l is the number of edges of a polyhedral convex cone; $k_l := [k_{l1}, \ldots, k_{lr_l}]^T \in \mathbb{R}^{r_l}, \ k_{lm} \geq 0 \ (m = 1, \ldots, r_l)$.

The relationship between the contact force applied on P_{ik}: f_{ik} and joint torques of the i-th finger can be expressed by

$$J_{ik}^T f_{ik} = \tau_i, \qquad (5)$$

$$\tau_i := [\tau_{i1}, \ldots, \tau_{iL_i}] \in \mathbb{R}^{L_i}, \qquad (6)$$

where $J_{ik} \in \mathbb{R}^{3 \times L_i}$ denotes the Jacobian matrix corresponding to the k-th contact point of the i-th finger; $\tau_{ij} \in \mathbb{R}^1$ denotes the joint torque of the j-th joint of the the i-th finger.

We determine Jacobian matrices and joint torques of each finger as follows.

fingers in position-controlled
Since a position-controlled finger is equivalent to the environment, $L_i = 0$; $J_i = [\ ] \in \mathbb{R}^{3 \times 0}$; and $\tau_i = [\ ] \in \mathbb{R}^0$. As a matter of formulation, J_i and τ_i is empty matrices [12].

fingers in force-controlled
Jacobian matrices and joint torques of force-controlled fingers are determined based on the fingers configuration.

fingers in hybrid control
Since a finger in hybrid force/position control is equivalent to a prismatic joint, $M_i = 1$; $L_i = 1$; $J_i = J_{i1} = n_{i1}$; $\tau_i = \tau_{i1} = f_{\mathrm{com},i1}$, where, $n_{ik} \in \mathbb{R}^3$ denotes a unit normal vector toward the object at P_{ik}; $f_{\mathrm{com},i1} \ (\geq 0)$ is a commanded force of the i-th finger along n_{i1}.

Here we define the following matrices:

$$W := \begin{bmatrix} I_3 & \cdots & I_3, \\ p_1 \times I_3 & \cdots & p_M \times I_3 \end{bmatrix} \in \mathbb{R}^{6 \times 3M}, \qquad (7)$$

$$C := \mathrm{diag}\,(C_1, \ldots, C_M) \in \mathbb{R}^{3M \times r}, \qquad (8)$$

$$k := [k_1^T, \ldots, k_M^T]^T \in \mathbb{R}^r, \qquad (9)$$

$$T := \mathrm{diag}\,(T_1, \ldots, T_M) \in \mathbb{R}^{3M \times 2M}, \qquad (10)$$

$$T_l := [t_{l1} \ t_{l2}] \in \mathbb{R}^{3 \times 2}, \qquad (11)$$

$$J := \mathrm{diag}\,(J_1, \ldots, J_N) \in \mathbb{R}^{3M \times L}, \qquad (12)$$

$$J_i := [J_{i1}^T, \ \ldots, \ J_{iM_i}^T]^T \in \mathbb{R}^{3M_i \times L_i}, \qquad (13)$$

$$f := [f_1^T, \ldots, f_M^T]^T \in \mathbb{R}^{3M}, \qquad (14)$$

$$\tau := [\tau_1^T, \ldots, \tau_N^T]^T \in \mathbb{R}^L, \qquad (15)$$

where I_n is an $n \times n$ identity matrix; $p_l \times I_3 \in \mathbb{R}^{3 \times 3}$ denotes a skew-symmetric matrix defined such that $(p_l \times I_3)x \equiv p_l \times x$; $t_{l1}, t_{l2} \in \mathbb{R}^3$ are unit tangential vectors at P_l defined such that $t_{l1}^T t_{l2} = 0$; diag(...) represents a block diagonal matrix.

All the contact forces can be expressed from (3) as:

$$f = Ck. \qquad (16)$$

All the frictional components of contact forces are:

$$T^T f = T^T Ck \in \mathbb{R}^{2M}. \qquad (17)$$

The relationship between the contact forces and the joint torques, (5) can be unified as follows:

$$h J^T f = J^T Ck = \tau. \qquad (18)$$

The equilibrium equation of the object in quasi-static manipulation can be expressed as follows:

$$Wf = WCk = -w_{\mathrm{ext}}, \qquad (19)$$

where $w_{\mathrm{ext}} \in \mathbb{R}^6$ is an external force and moment applied to the object such as gravity.

III. Constraint on Static Friction

A. Kinematic Constraint of Contact Points

Static friction is, in physical phenomena, caused by shear strain on microscopic partially-adhered contact points [13]. In general, static friction in rigid-body model is expressed with normal component of reaction on the contact points and Coulomb's friction law such as (17). Although modeling of static friction is variously studied, not only with rigid-body and Coulomb's law but also with elastic contacts [14], [15], [16], rigid-body model is often used in the viewpoint of a facility for analysis with less parameters.

We note that static frictional forces can be applied only in the opposite direction of microscopic tangential movements of contact points, which correspond to the shear strain mentioned above. Although the movements do not actually occur in static analysis, the movements has to satisfy the constraint on kinematics of the rigid-body object [5], [6]. Omata et al. study static indeterminate contact forces in power grasps and formulate the constraint of tangential movements of contact points. We modified them from the original formulations to apply the constraint to quasi-static graspless manipulation [3], and also applied the modified one to static grasping problems [17]. Our modified constraint on static friction in quasi-static problems does not work well, however, in some cases in which actual sliding and kinetic friction occur. Thus we modify the constraint again to apply it to quasi-static manipulation appropriately in this paper.

B. Introduction of Virtual Sliding

Let us suppose that a object is on the floor and the surface contact can be approximated by finite contact points. As mentioned above, static friction is actually caused by shear strain of adhered contact points, and it can be expressed as Coulomb's friction in rigid-body model as (17). Because static friction in physical phenomena is applied only to the opposite direction of the shear strain, the direction of the Coulomb's friction is also constrained. Let us suppose "*virtual*" infinitesimal tangential displacements of contact points to represent the opposite direction of Coulomb's friction in rigid-body model. In other words, the virtual tangential displacements in rigid-body model correspond to the shear strain of the contact points, which causes static friction in physical phenomena. We call the displacement *virtual sliding*. Note that virtual sliding, which causes static frictional force, must be distinguished from *actual sliding*, which causes kinetic frictional force.

We define a selection matrix [3], [17]:

$$\boldsymbol{B} := \operatorname{diag}(b_1 \boldsymbol{I}_3, \ldots, b_M \boldsymbol{I}_3) \in \mathbb{R}^{3M \times 3M} \quad (20)$$

$$b_l := \begin{cases} 1 & \text{(when a virtual sliding exists at } P_l \text{,)} \\ 0 & \text{(otherwise.)} \end{cases} \quad (21)$$

At every contact point selected by $\boldsymbol{B}$, only virtual sliding is led by a virtual motion of the object, and then it must satisfy the following constraint [3]:

$$\boldsymbol{B} \begin{bmatrix} \boldsymbol{W}^T & \boldsymbol{J} \end{bmatrix} \begin{bmatrix} \boldsymbol{V} \\ -\dot{\boldsymbol{\theta}} \end{bmatrix} = \boldsymbol{T} \dot{\boldsymbol{Y}}, \quad (22)$$

where

$$\dot{\boldsymbol{Y}} := \begin{bmatrix} \dot{\boldsymbol{Y}}_1^T, \ldots, \dot{\boldsymbol{Y}}_M^T \end{bmatrix}^T \in \mathbb{R}^{2M}, \quad (23)$$

$$\dot{\boldsymbol{Y}}_l := \begin{bmatrix} \dot{Y}_{l1}, & \dot{Y}_{l2} \end{bmatrix}^T \in \mathbb{R}^2, \quad (24)$$

$$\boldsymbol{V} := [\boldsymbol{v}_0^T, \boldsymbol{\omega}_0^T]^T \in \mathbb{R}^6, \quad (25)$$

$$\dot{\boldsymbol{\theta}} := \begin{bmatrix} \dot{\theta}_{11}, \dot{\theta}_{12}, \ldots, \dot{\theta}_{NL_N} \end{bmatrix}^T \in \mathbb{R}^L; \quad (26)$$

$\dot{Y}_{l1}, \dot{Y}_{l2}$ are virtual sliding velocity toward basis vectors:$\boldsymbol{t}_{l1}, \boldsymbol{t}_{l2}$ at a contact point, P_l, respectively; $\boldsymbol{v}_0 \in \mathbb{R}^3$ is the virtual velocity of the object; $\boldsymbol{\omega}_0 \in \mathbb{R}^3$ is the virtual angular velocity of the object; $\dot{\theta}_{ij}$ is the virtual joint velocity of the j-th joint of the i-th finger. Accordingly only virtual sliding that satisfies (22) is feasible. In other words, $\dot{\boldsymbol{Y}}$ is feasible when a series of $\boldsymbol{V}, \dot{\boldsymbol{\theta}}$ and $\boldsymbol{B}$ that satisfies (22) is present. Since static friction can occur only in the opposite direction to feasible $\dot{\boldsymbol{Y}}$, it is constrained by (22).

C. Treatment of Actual Sliding

Let us apply the kinematic constraint on static friction expressed as (22) to quasi-static graspless manipulation. In graspless manipulation such as pushing, the manipulated object moves quasi-statically, and there are some contact points in actual sliding, which causes kinetic friction force. Thus it is important to apply (22) appropriately depending on each sliding mode of contact point. How is the constraint applied to contact points in actual sliding?

In [3], (22) is not applied to the contact points in actual sliding, that is, $b_l = 0$, when $d_l = 1$. The ignorance brings, however, unreasonable results in some cases. Let us consider a pushing operation such as Fig. 2, where the contact points between the object and the floor are in actual sliding, and kinetic frictional force is present in the direction of $+x$. If (22) is applied to only the contact points between the robot and object, it will allows virtual slidings in the same direction at the contact points, according to a virtual motion of the both rigid bodies (Fig. 2 (a)). Thus, virtual slidings in the direction of $+z$ are also allowed to occur and cause static frictional forces in the direction of $-z$ (Fig. 2 (b)). In this case, the resultant force of the static friction and the kinetic friction may reach an infinity value. If we replace the actual sliding with virtual sliding in the case, the combination of virtual sliding will not satisfy (22). Unreasonable resultant frictional forces like this are reported in [17], where both are static friction.

To avoid the unreasonable results, we assume that virtual sliding occurs at every contact point in actual sliding, that is, $b_l = 1$, when $d_l = 1$, to apply the kinematic constraint on static friction expressed as (22) to the contact points. It means the infinitesimal tangential displacements on the contact points in actual sliding always occur because the kinetic friction that prevent the displacements is applied at same time. Note that virtual sliding at an actually sliding contact point does not cause *static* friction. Consequently, existence of virtual sliding on the contact points means that the infinitesimal tangential displacement at the contact point is present, and then static or kinetic friction occurs there.
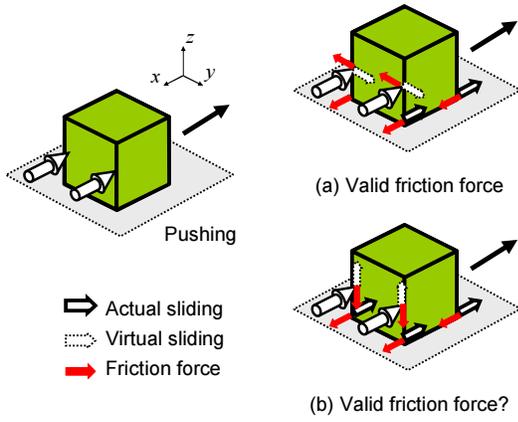
Fig. 2. Test of virtual sliding ignoring actual sliding
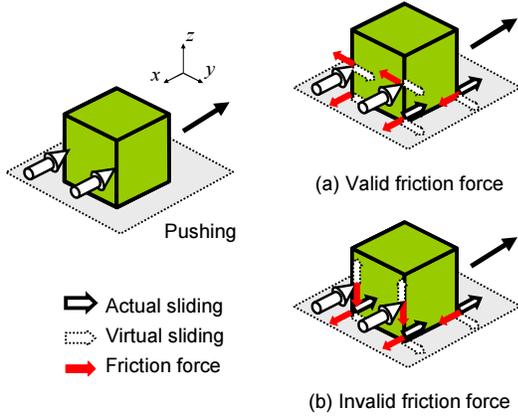
Fig. 3. Test of virtual sliding considering actual sliding

When virtual slidings at actually sliding contact points are considered, invalid combinations of virtual slidings such as Fig. 3 (b) is excluded by the kinematic constraint of (22). On the other hand, combinations such as Fig. 3 (a) satisfies (22).

IV. ANALYSIS OF QUASI-STATIC CONTACT FORCES

As mentioned above, frictional forces cannot exist at the contact points that are not selected by $\boldsymbol{B}$ (and $\boldsymbol{D}$). The constraint can be written as:

$$\boldsymbol{T}^T (\boldsymbol{I}_{3M} - \boldsymbol{B}) \boldsymbol{C} \boldsymbol{k} = \boldsymbol{0}. \tag{27}$$

Let us consider the constraint on static friction that static frictional forces occur to prevent virtual slidings that satisfies (22). As [3], [17], we define the following matrix to represent the combination of the signs of virtual sliding velocity, $\dot{\boldsymbol{Y}}$:

$$\boldsymbol{S} := \mathrm{diag}\,(s_{11}, s_{12}, s_{21}, s_{22}, \ldots, s_{M1}, s_{M2}) \in \mathbb{R}^{2M \times 2M}, \tag{28}$$

$$s_{lm} := \begin{cases} +1 & (b_l = 1 \text{ and } \dot{Y}_{lm} > 0) \\ -1 & (b_l = 1 \text{ and } \dot{Y}_{lm} < 0) \\ 0 & (b_l = 0). \end{cases} \tag{29}$$

Thus,

$$\dot{\boldsymbol{Y}} = \boldsymbol{S} \boldsymbol{q}. \tag{30}$$

where $\boldsymbol{q}\left(\in \mathbb{R}^{2M}\right) > \boldsymbol{0}$. When $\boldsymbol{S}$ and $\boldsymbol{q}$ satisfy (22) and (30), the constraint on static friction can be expressed as:

$$\boldsymbol{S} \boldsymbol{T}^T (\boldsymbol{B} - \boldsymbol{D}) \boldsymbol{C} \boldsymbol{k} \leq \boldsymbol{0} \tag{31}$$

From (18), (19), (27) and (31), we can calculate a set of possible contact forces for a subcase specified by $\boldsymbol{S}$ by solving the following equations and inequalities:

$$\begin{cases} \boldsymbol{W} \boldsymbol{C} \boldsymbol{k} = -\boldsymbol{w}_{\mathrm{ext}} \\ \boldsymbol{J}^T \boldsymbol{C} \boldsymbol{k} = \boldsymbol{\tau} \\ \boldsymbol{T}^T (\boldsymbol{I}_{3M} - \boldsymbol{B}) \boldsymbol{C} \boldsymbol{k} = \boldsymbol{0} \\ \boldsymbol{S} \boldsymbol{T}^T (\boldsymbol{B} - \boldsymbol{D}) \boldsymbol{C} \boldsymbol{k} \leq \boldsymbol{0} \\ \boldsymbol{k} \geq \boldsymbol{0}, \end{cases} \tag{32}$$

Consequently, we can consider various patterns of the directions of reaction forces caused by various external wrenches, $\boldsymbol{w}_{\mathrm{ext}}$, by changing $\boldsymbol{S}$, which satisfies (22) and (30).

V. JOINT TORQUE OPTIMIZATION

Algorithms of joint torque optimization for power grasps are studied based on maximizing the margin between a friction cone and contact force in it [18]. The margin contributes the robustness against changes of disturbances because a contact breaks when the corresponding margin becomes zero. There are two algorithms: *Algorithm 1* is maximizing joint torques to resist against as large changes of disturbances as possible; *Algorithm 2* is minimizing joint torques that equilibrates contact forces with given disturbances as [10].

On the fact that indeterminate contact forces in manipulation are bounded even when the constraint on static friction is not considered, the minimum margin between the vertex of the range of the indeterminate contact forces and its corresponding friction cone is maximized [18]. On the other hand, our proposed analysis of indeterminate contact forces with the modified constraint on static friction shows unboundedness of the contact forces in some cases [17], and then, the minimum margin is always zero. Thus, our proposed optimization of joint torques maximizes the margin between each friction cone and contact force [4]. Note that the proposed algorithms expect the best selection in the indeterminate contact forces, and it is not always applied.

We improve our previous algorithms [4] to apply them to quasi-static graspless manipulation with the proposed analysis in this paper. Our previous method can be applied only to static non-grasping scenes because of the overestimation of static friction mentioned in Sec. III-C.

We define $e\ (\geq 0)$ as a margin between a friction cone and contact force within it (Fig. 4). When a friction cone is approximated by a polyhedral convex cone (4), e can be considered as the following two:

- a margin between contact force and each side face,
- a margin between contact force and the base face,

and it must satisfy:

$$e + \boldsymbol{n}_l^T \boldsymbol{C}_l \boldsymbol{k}_l \leq f_{n,\max}, \tag{33}$$

$$e \leq \boldsymbol{n}_l^T \boldsymbol{C}_l \boldsymbol{k}_l, \tag{34}$$

$$e - \boldsymbol{h}_{lm}^T \boldsymbol{C}_l \boldsymbol{k}_l \leq 0, \tag{35}$$
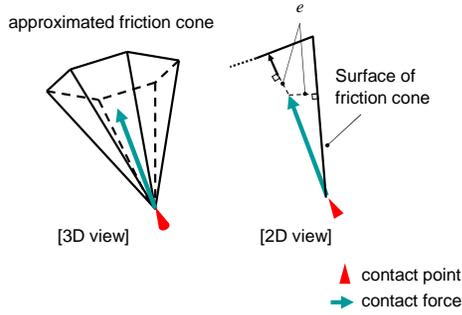
Fig. 4. Margins between a friction cone and contact force

where $\boldsymbol{n}_l \in \mathbb{R}^3$ is a unit normal vector toward inside of the object at P_l; $\boldsymbol{h}_{lm} \in \mathbb{R}^3$ is a unit normal vector toward outside of the m-th side face of the polyhedral cone at P_l; $f_{n,\max}$ denotes the height of friction cones, that is, the limitation of the normal component of contact forces. Here we rewrite (33), (34) and (35) as:

$$e\mathbf{1}_M \le \boldsymbol{N}^T \boldsymbol{C}\boldsymbol{k} \le f_{n,\max}\mathbf{1}_M - e\mathbf{1}_M, \tag{36}$$

$$e\mathbf{1}_{rM} - \boldsymbol{H}^T \boldsymbol{C}\boldsymbol{k} \le \mathbf{0}, \tag{37}$$

where,

$$\boldsymbol{H}_l := \begin{bmatrix} \boldsymbol{h}_{l1} & \dots & \boldsymbol{h}_{lr_l} \end{bmatrix} \in \mathbb{R}^{3 \times r_l}, \tag{38}$$

$$\boldsymbol{H} := \operatorname{diag}(\boldsymbol{H}_1, \dots, \boldsymbol{H}_M) \in \mathbb{R}^{3M \times rM}, \tag{39}$$

$$\boldsymbol{N} := \operatorname{diag}(\boldsymbol{n}_1, \dots, \boldsymbol{n}_M) \in \mathbb{R}^{3M \times M}, \tag{40}$$

$$\mathbf{1}_n := [1, 1, \dots, 1] \in \mathbb{R}^n. \tag{41}$$

We cannot define margins at the contact points in actual sliding ($d_l = 1$) because each contact force there always occurs along the edge of polyhedral friction cone. Thus, the contact points are excluded as follows:

$$(\boldsymbol{I}_{rM} - \boldsymbol{D}^*)(e\mathbf{1}_{rM} - \boldsymbol{H}^T \boldsymbol{C}\boldsymbol{k}) \le \mathbf{0}, \tag{42}$$

$$\boldsymbol{D}^* := \operatorname{diag}(d_1 \boldsymbol{I}_{r_1}, \dots, d_M \boldsymbol{I}_{r_M}) \in \mathbb{R}^{rM \times rM}. \tag{43}$$

With the modified constraint on static friction (Sec. III), *Algorithm 1* can be written as:

$$\max e$$

$$\text{subject to} \begin{cases} \boldsymbol{WC}\boldsymbol{k} = -\boldsymbol{w}_{\text{ext}}, \\ \boldsymbol{J}^T \boldsymbol{C}\boldsymbol{k} = \boldsymbol{\tau}, \\ \boldsymbol{T}^T (\boldsymbol{I}_{3M} - \boldsymbol{B}) \boldsymbol{C}\boldsymbol{k} = \mathbf{0}, \\ \boldsymbol{S}\boldsymbol{T}^T (\boldsymbol{B} - \boldsymbol{D}) \boldsymbol{C}\boldsymbol{k} \le \mathbf{0}, \\ e\mathbf{1}_M \le \boldsymbol{N}^T \boldsymbol{C}\boldsymbol{k} \le f_{n,\max}\mathbf{1}_M - e\mathbf{1}_M, \\ (\boldsymbol{I}_{rM} - \boldsymbol{D}^*)(e\mathbf{1}_{rM} - \boldsymbol{H}^T \boldsymbol{C}\boldsymbol{k}) \le \mathbf{0}, \\ \boldsymbol{k} \ge \mathbf{0}, \\ \boldsymbol{\tau}_{\min} \le \boldsymbol{\tau} \le \boldsymbol{\tau}_{\max}, \end{cases} \tag{44}$$

where $\boldsymbol{\tau}_{\min}$ and $\boldsymbol{\tau}_{\max}$ denote the minimum and maximum bounds of joint torques respectively. *Algorithm 1* tends to calculate excessive joint torques for robust contacts against as large changes of disturbances as possible.

When we require a margin, e_0, *Algorithm 2* can be written as:

$$\min \tau_{\max}$$

$$\text{subject to} \begin{cases} \boldsymbol{WC}\boldsymbol{k} = -\boldsymbol{w}_{\text{ext}}, \\ \boldsymbol{J}^T \boldsymbol{C}\boldsymbol{k} = \boldsymbol{\tau}, \\ \boldsymbol{T}^T (\boldsymbol{I}_{3M} - \boldsymbol{B}) \boldsymbol{C}\boldsymbol{k} = \mathbf{0}, \\ \boldsymbol{S}\boldsymbol{T}^T (\boldsymbol{B} - \boldsymbol{D}) \boldsymbol{C}\boldsymbol{k} \le \mathbf{0}, \\ e_0 \mathbf{1}_M \le \boldsymbol{N}^T \boldsymbol{C}\boldsymbol{k} \le f_{n,\max}\mathbf{1}_M - e_0\mathbf{1}_M, \\ (\boldsymbol{I}_{rM} - \boldsymbol{D}^*)(e_0\mathbf{1}_{rM} - \boldsymbol{H}^T \boldsymbol{C}\boldsymbol{k}) \le \mathbf{0}, \\ \boldsymbol{k} \ge \mathbf{0}, \\ -\tau_{\max}\mathbf{1}_L \le \boldsymbol{\tau} \le -\tau_{\max}\mathbf{1}_L. \end{cases} \tag{45}$$

VI. NUMERICAL EXAMPLES

We assume that the manipulated object is a polyhedron whose mass is 1.0 and mass distribution is uniform; the gravitational acceleration is 9.8; The coefficients of both static and kinetic friction are equal; each friction cone is approximated by a regular polyhedral convex cone with 32 edges; The limitation of each joint torque, τ_l, is $-10.0 \le \tau_l \le 10.0$; the given robustness in *Algorithm 2*, $e_0 = 0.1$; the height of each friction cone, $f_{n,\max} = 50$. The known external wrench, $\boldsymbol{w}_{\text{ext}}$, applies only gravitational force; the origin of the coordinate is the center of the object. The computation times in this paper are measured on a Linux PC with Core2 Quad–2.66 GHz.

A. Non-grasp Manipulation

Let us consider the scene where robotic fingers support a spherical object with the radius of 0.1 on the slope of 45 [deg] (Fig. 5); The parameters for the calculation are as follows:

$$\boldsymbol{p}_1 = \begin{bmatrix} 0.0612 \\ -0.0707 \\ -0.0354 \end{bmatrix}, \boldsymbol{p}_2 = \begin{bmatrix} -0.0707 \\ 0.0 \\ -0.0707 \end{bmatrix}, \boldsymbol{p}_3 = \begin{bmatrix} 0.0612 \\ -0.0707 \\ -0.0354 \end{bmatrix}$$

$$\boldsymbol{T}_1 = \begin{bmatrix} -0.5 & -0.612 \\ 0.0 & -0.707 \\ -0.866 & 0.354 \end{bmatrix}, \boldsymbol{T}_2 = \begin{bmatrix} -0.707 & 0.0 \\ 0.0 & -1.0 \\ 0.707 & 0.0 \end{bmatrix}, \boldsymbol{T}_3 = \begin{bmatrix} -0.5 & 0.612 \\ 0.0 & -0.707 \\ -0.866 & -0.354 \end{bmatrix}$$

$$\boldsymbol{n}_1 = \begin{bmatrix} -0.612 \\ 0.707 \\ 0.354 \end{bmatrix}, \boldsymbol{n}_2 = \begin{bmatrix} 0.707 \\ 0.0 \\ 0.707 \end{bmatrix}, \boldsymbol{n}_3 = \begin{bmatrix} -0.612 \\ -0.707 \\ 0.354 \end{bmatrix}$$

$$\boldsymbol{J} = \begin{bmatrix} 0.0919 & -0.53 & -0.106 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 \\ 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0 & 0.0919 & -0.53 & 0.106 \end{bmatrix}^T$$

We can obtain the following calculation results with *Algorithm 1*: $e = 1.45$, $\boldsymbol{\tau} = [-0.76, -0.76]^T$. and with *Algorithm 2*, $e_0 = 0.1$ (constant), $\boldsymbol{\tau} = [-0.52, -0.52]^T$ The computation times in the cases are 0.09 and 0.24 CPU seconds respectively. Note that the calculated contact forces are not always applied because they are the best selection of indeterminate contact forces estimated with the variable combinations of $\boldsymbol{B}$ and $\boldsymbol{S}$.

B. Quasi-static Manipulation

Let us consider the case where a jaw with two robotic fingers pinches a cuboid with the size of $0.2 \times 0.2 \times 0.2$ on the slope with 45 [deg] and slides it along the slope (Fig. 6).
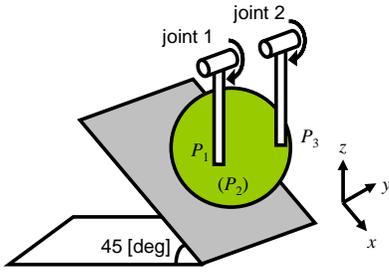
Fig. 5. Supporting a spherical object on the slope with two fingers
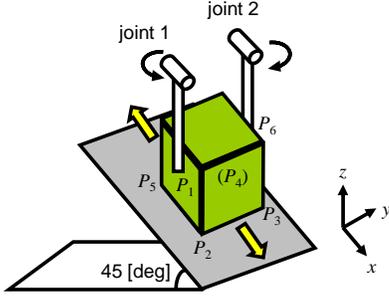
Fig. 6. Sliding a cuboid with pinching

The parameters for the calculation are as follows:

$$\boldsymbol{p}_1 = \begin{bmatrix} 0.035 \\ -0.1 \\ 0.035 \end{bmatrix}, \ \boldsymbol{p}_2 = \begin{bmatrix} 0.0 \\ -0.1 \\ -0.014 \end{bmatrix}, \ \boldsymbol{p}_3 = \begin{bmatrix} 0.0 \\ 0.1 \\ -0.014 \end{bmatrix},$$

$$\boldsymbol{p}_4 = \begin{bmatrix} -0.014 \\ 0.1 \\ 0.0 \end{bmatrix}, \ \boldsymbol{p}_5 = \begin{bmatrix} -0.014 \\ -0.1 \\ 0.0 \end{bmatrix}, \ \boldsymbol{p}_6 = \begin{bmatrix} 0.035 \\ 0.1 \\ 0.035 \end{bmatrix},$$

$$\boldsymbol{T}_1 = \begin{bmatrix} -\sqrt{2}/2 & -\sqrt{2}/2 \\ 0.0 & 0.0 \\ \sqrt{2}/2 & -\sqrt{2}/2 \end{bmatrix}, \ \boldsymbol{T}_2 = \begin{bmatrix} -\sqrt{2}/2 & 0.0 \\ \sqrt{2}/2 & 0.0 \\ 0.0 & -1.0 \end{bmatrix}, \ \boldsymbol{T}_3 = \begin{bmatrix} -\sqrt{2}/2 & 0.0 \\ \sqrt{2}/2 & 0.0 \\ 0.0 & -1.0 \end{bmatrix},$$

$$\boldsymbol{T}_4 = \begin{bmatrix} -\sqrt{2}/2 & 0.0 \\ \sqrt{2}/2 & 0.0 \\ 0.0 & -1.0 \end{bmatrix}, \ \boldsymbol{T}_5 = \begin{bmatrix} -\sqrt{2}/2 & 0.0 \\ \sqrt{2}/2 & 0.0 \\ 0.0 & -1.0 \end{bmatrix}, \ \boldsymbol{T}_6 = \begin{bmatrix} -\sqrt{2}/2 & \sqrt{2}/2 \\ 0.0 & 0.0 \\ \sqrt{2}/2 & \sqrt{2}/2 \end{bmatrix},$$

$$\boldsymbol{n}_1 = \begin{bmatrix} 0.0 \\ 0.0 \\ -1.0 \end{bmatrix}, \ \boldsymbol{n}_2 = \boldsymbol{n}_3 = \boldsymbol{n}_4 = \boldsymbol{n}_5 = \begin{bmatrix} \sqrt{2}/2 \\ \sqrt{2}/2 \\ 0.0 \end{bmatrix}, \ \boldsymbol{n}_6 = \begin{bmatrix} 0.0 \\ 0.0 \\ 1.0 \end{bmatrix},$$

$$\boldsymbol{J}_{11} = \begin{bmatrix} 0.0 & 0.0 \\ -0.1 & 0.0 \\ 0.0 & 0.0 \end{bmatrix}, \ \boldsymbol{J}_{21} = \begin{bmatrix} 0.0 & 0.0 \\ 0.0 & -0.1 \\ 0.0 & 0.0 \end{bmatrix}$$

First, when the jaw keeps the object in stationary on the slope, *Algorithm 1* calculates the robustness of the object and the joint torques in this case as: $e = 1.65$, $\boldsymbol{\tau} = [-4.83, 4.83]^T$. Then *Algorithm 2* calculates them as: $e_0 = 0.1$ (const.), $\boldsymbol{\tau} = [-0.96, 0.96]^T$. The calculation times are 1.2 and 1.4 CPU seconds respectively.

Next, when the robot fingers slide the object up along the slope, the values calculated by *Algorithm 1* are $e = 2.30$, $\boldsymbol{\tau} = [-4.76, 4.76]^T$, and those by *Algorithm 2* are $e_0 = 0.1$ (const.), $\boldsymbol{\tau} = [-1.56, 1.56]^T$. The calculation times are 0.07 and 0.11 CPU seconds respectively.

Third, when the robot fingers slide the object down along the slope, the values calculated by *Algorithm 1* are $e = 3.98$, $\boldsymbol{\tau} = [-4.60, 4.60]^T$, and those by *Algorithm 2* are $e_0 = 0.1$ (const.), $\boldsymbol{\tau} = [-0.85, 0.85]^T$. The calculation times are 0.08 and 0.12 CPU seconds respectively.

VII. Conclusions and Future Works

In this paper, we modified the analysis of contact forces in quasi-static graspless manipulation presented in [3], and improved joint torque optimization for grasp/graspless manipulation presented in [4]. Overestimation of contact forces caused by an inappropriate application of the constraint on static friction was resolved by an appropriate treatment of contact points where kinetic friction occurs. Additionally, improved joint torque optimization based on the analysis can deal with the cases of not only static grasp/graspless manipulation but also quasi-static graspless manipulation.

As future works, reduction of computational cost should be investigated as [17]. In addition, the joint torque optimization can be applied to planning of graspless manipulation.

References

[1] Y. Aiyama, M. Inaba, and H. Inoue, "Pivoting: A new method of graspless manipulation of object by robot fingers," in *Proc. of the 1993 IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, jul 1993, pp. 136–143.

[2] M. T. Mason, "Progress in nonprehensile manipulation," *Int. J. of Robotics Research*, vol. 18, no. 11, pp. 1129–1141, November 1999.

[3] Y. Maeda and S. Makita, "A quantitative test for the robustness of graspless manipulation," in *Proc. of IEEE Int. Conf. on Robotics and Automation*, Orlando, FL, U.S.A., May 2006, pp. 1743–1748.

[4] S. Makita, S. Nakamura, and Y. Maeda, "Joint torque optimization for grasp/graspless manipulation," in *The 3rd International Symposium on Measurement, Analysis and Modeling of Human Functions*, I. Fragoso, F. Carnide, and F. Vieira, Eds. Lisbon, Portugal: Edições Faculdade de Motricidade Humana, 2007, pp. 203–210.

[5] T. Omata and K. Nagata, "Rigid body analysis of the indeterminate grasp force in power grasps," *IEEE Trans. on Robotics and Automation*, vol. 16, no. 1, pp. 46–54, February 2000.

[6] T. Omata, "Rigid body analysis of power grasps: Bounds of the indeterminate grasp force," in *Proc. of IEEE Int. Conf. on Robotics and Automation*, Seoul, Korea, May 2001, pp. 2203–2209.

[7] J. C. Trinkle, J. M. Abel, and R. P. Paul, "An investigation of frictionless enveloping grasping in the plane," *Int. J. of Robotics Research*, vol. 7, no. 3, pp. 33–51, June 1988.

[8] A. Bicchi, "Analysis and control of powre grasping," in *Proc. of IEEE Int. Workshop on Intelligent Robots and Systems*, 1991, pp. 691–697.

[9] X. Y. Zhang, Y. Nakamura, K. Goda, and K. Yoshimoto, "Robustness of power grasp," in *Proc. of IEEE Int. Conf. on Robotics and Automation*, May 1994, pp. 2828–2835.

[10] Y. Yong, K. Takeuchi, and T. Yoshikawa, "Optimization of robot hand power grasps," in *Proc. of IEEE Int. Conf. on Robotics and Automation*, Leuven, Belgium, May 1998, pp. 3341–3347.

[11] S. Hirai and H. Asada, "Kinematics and statics of manipulation using the theory of polyhedral convex cones," *Int. J. of Robotics Research*, vol. 12, no. 5, pp. 434–447, 1993.

[12] C. de Boor, "An empty exercise," *ACM SIGNUM Newsletter*, vol. 25, no. 4, pp. 2–6, 1990.

[13] P. J. Blau, *Friction Science and Technology*, 1st ed. CRC Press, 1996.

[14] K. B. Shimoga, "Robot grasp synthesis algorithm: A survey," *Int. J. of Robotics Research*, vol. 15, no. 3, pp. 230–266, June 1996.

[15] A. Bicchi and V. Kumar, "Robotic grasping and contact: A review," in *Proc. of IEEE Int. Conf. on Robotics and Automation*, San Francisco, CA, U.S.A., April 2000, pp. 348–353.

[16] E. Rimon, J. W. Burdick, and T. Omata, "A polyhedral bound on the indeterminate contact forces in planar quasi-rigid fixturing and grasping arrangements," *IEEE Trans. on Robotics*, vol. 22, no. 2, pp. 240–255, April 2006.

[17] Y. Maeda, K. Oda, and S. Makita, "Analysis of indeterminate contact forces in robotic grasping and contact tasks," in *Proc. of IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, San Diego, CA, U.S.A., 2007, pp. 1570–1575.

[18] T. Omata, "Algorithms for computing optimal joint torques for power grasps," *Trans. of the Japan Society of Mechanical Engineers, C (in Japanese)*, vol. 68, no. 672, pp. 2395–2401, August 2002.