Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2013, Article ID 681710, 9 pages
http://dx.doi.org/10.1155/2013/681710

Research Article

Optimal Grasping Manipulation for Multifingered Robots Using Semismooth Newton Method

Chun-Hsu Ko[1] and Jein-Shan Chen[2,3]

[1] *Department of Electrical Engineering, I-Shou University, Kaohsiung 84001, Taiwan*
[2] *Department of Mathematics, National Taiwan Normal University, Taipei 11677, Taiwan*
[3] *Mathematics Division, National Center for Theoretical Sciences, Taipei, Taiwan*

Correspondence should be addressed to Jein-Shan Chen; jschen@math.ntnu.edu.tw

Received 6 July 2013; Accepted 11 September 2013

Academic Editor: Masoud Hajarian

Multifingered robots play an important role in manipulation applications. They can grasp various shaped objects to perform point-to-point movement. It is important to plan the motion path of the object and appropriately control the grasping forces for multifingered robot manipulation. In this paper, we perform the optimal grasping control to find both optimal motion path of the object and minimum grasping forces in the manipulation. The rigid body dynamics of the object and the grasping forces subjected to the second-order cone (SOC) constraints are considered in optimal control problem. The minimum principle is applied to obtain the system equalities and the SOC complementarity problems. The SOC complementarity problems are further recast as the equations with the Fischer-Burmeister (FB) function. Since the FB function is semismooth, the semismooth Newton method with the generalized Jacobian of FB function is used to solve the nonlinear equations. The 2D and 3D simulations of grasping manipulation are performed to demonstrate the effectiveness of the proposed approach.

1. Introduction

Multifingered robots have attracted much attention in robotics manipulation applications. They can grasp various shaped objects and dexterously perform point-to-point manipulations. Many researches [1–6] have been proposed for grasping and manipulating objects with multifingered robots. Miller and Allen [2] proposed a user interface with grasp quality evaluation for the robot hand design. Yokokohji et al. [6] proposed a measure of dynamic manipulability of multifingered grasping for the systems consisting of a multifingered hand and a grasped object. Xu and Li [5] proposed a modeling method for the manipulation involving finger gaits. Kawamura et al. [1] used soft finger tips for stable grasping. Takahashi et al. [4] proposed robust force and position control with the information of tactile sensor. It is important to appropriately control the grasping forces for multifingered robot manipulation.

Since the grasping manipulation utilizes the contact and friction forces to hold and move an object, the grasping forces should satisfy the point-contact friction constraint and be equal to the dynamic wrench of the grasped object. It is required to find the minimum forces for moving the grasped object in the manipulation. Boyd and Wegbreit [7] used the semidefinite programming and second-order cone programming to efficiently find the grasping forces. Helmke et al. [8] proposed quadratically convergent algorithms for optimal dexterous grasping. Han et al. [9] used the convex optimization involving linear matrix inequalities for grasping forces computation. Liu et al. [10] presented a unified geometric framework for efficient grasping force optimization. Zheng et al. [11] developed an algorithm to determine the minimum required friction coefficient and the corresponding reliable minimum contact forces in practice. Ko et al. [12] proposed a neural network to calculate the optimal grasping forces. Because the external wrench of the object varies with the manipulation path and orientation of the object, it is important to plan a manipulation trajectory [13] for achieving the minimum grasping forces.

In this paper, we perform the optimal grasping control to find both optimal manipulation path of the object and minimum grasping forces. The rigid body dynamics of the object and the grasping forces subjected to the second-order cone (SOC) constraints are considered in the grasping control problem. The minimum principle [14] is applied to obtain the system equalities and the SOC complementarity problems. The SOC complementarity problems can be recast as the equations with the Fischer-Burmeister (FB) function. The semismooth Newton method with the generalized Jacobian of FB function is then used to solve the equations. Finally, simulations of optimal grasping manipulation are performed to demonstrate the effectiveness of the proposed approach.

The remainder of this paper is organized as follows: Section 2 describes the optimal grasping control problem. In Section 3, the semismooth Newton method with the generalized Jacobian of Fischer-Burmeister function is addressed. Section 4 presents the simulation results of 2D and 3D grasping manipulations. Finally, concluding remarks are given in Section 5.

2. Optimal Grasping Control

Figure 1 shows the multifingered robot grasping manipulation. The multifingered robot grasps and moves the object from the initial position to the final position. The dynamic equation of the object can be expressed with Newton-Euler equations [15, 16] as

$$\begin{aligned} \dot{y} &= v, \\ \dot{v} &= \frac{1}{m} R G_1 u + \begin{bmatrix} 0 & 0 & -g \end{bmatrix}^T, \\ \dot{q} &= Q\omega, \\ \dot{\omega} &= I^{-1}\left(R G_2 u - \omega \times (I\omega)\right), \end{aligned} \tag{1}$$

where y is the position, v is the velocity, $q = \begin{bmatrix} q_1 & q_2 & q_3 \end{bmatrix}^T$ is the quaternion, ω is the angular velocity, m is the object mass, I is the matrix of moment of inertia, g is the gravity constant, u means the grasping forces which is represented by a matrix, $[G_1\ G_2]$ is the contact matrix, R is the rotation matrix of the object, and Q can be expressed as

$$Q = \frac{1}{2}\begin{bmatrix} q_0 & q_3 & -q_2 \\ -q_3 & q_0 & q_1 \\ q_2 & -q_1 & q_0 \end{bmatrix} \quad \text{with } q_0 = \sqrt{q_1^2 + q_2^2 + q_3^2}. \tag{2}$$

Moreover, the grasping forces are subject to the contact friction constraint, expressed as

$$\left\|\left(u_{i2}, u_{i3}\right)\right\| \le \mu u_{i1}, \tag{3}$$

where u_{i1} is the normal force of the ith finger, u_{i2} and u_{i3} are the friction forces of the ith finger, $\|\cdot\|$ is the 2-norm, and μ is the friction coefficient.

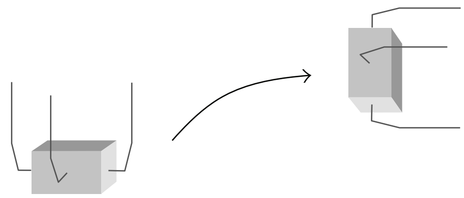

Figure 1: Multifingered robot manipulation.

To find the path that can be achieved with the minimum grasping forces, the optimal control problem can be recast as

$$\begin{aligned} \min \int_0^T & L\,dt, \\ \text{s.t.} \quad & \dot{x} = f(x, u), \\ & x(0) = x_0, \\ & x(T) = x_T, \\ & Du \in \mathcal{K}^d \times \mathcal{K}^d \times \cdots \times \mathcal{K}^d, \end{aligned} \tag{4}$$

where

$$L = \frac{u^T u}{2}, \qquad x = \begin{bmatrix} y \\ v \\ q \\ \omega \end{bmatrix}. \tag{5}$$

In addition, f represents the right hand side of system (1), T is the control duration, x_0 and x_T are the initial and final states, respectively, D is the diagonal matrix with the friction coefficient, and $\mathcal{K}$ denotes the second-order cone which is given by

$$\mathcal{K}^d := \left\{ \begin{bmatrix} z_1 \\ z_2 \end{bmatrix} \in \mathbb{R} \times \mathbb{R}^{d-1} \mid \|z_2\| \le z_1 \right\}. \tag{6}$$

The optimal control problem (4) can be solved by using the Pontryagin's minimum principle, see [14, 17, 18]. In optimization language, it is to write out the KKT conditions for problem (4) which consist of two parts. The first part involves a few equalities about Lagrange multipliers, while the other part is related to complementarity conditions. More specifically, with the Hamiltonian function, the first part can be reformulated as follows:

$$\begin{aligned} \dot{x} - H_\lambda &= \dot{x} - f(x, u) = 0, \\ \dot{\lambda} + H_x &= \dot{\lambda} + \lambda^T f_x = 0, \\ H_u &= L_u + \lambda^T f_u + \eta^T D = 0, \\ \phi(x(0), x(T)) &= 0, \\ \lambda(0) &+ \phi_{x(0)}^T \sigma = 0, \\ \lambda(T) &+ \phi_{x(T)}^T \sigma = 0, \end{aligned} \tag{7}$$

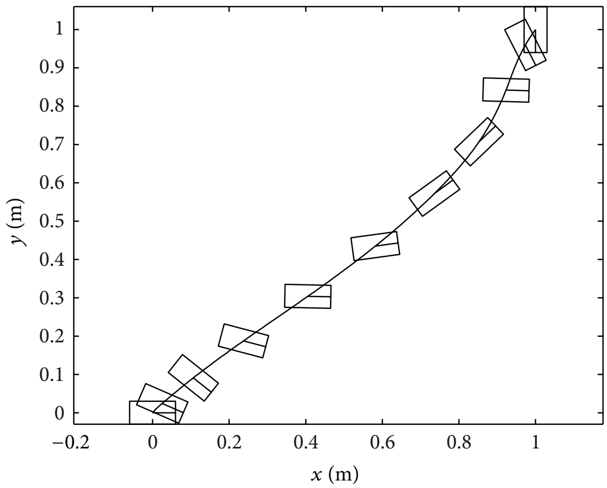

Figure 2: The manipulation path of 90 degrees rotation with $T = 1$ s and $\mu = 0.3$.

where λ, η, and σ are the Lagrange multipliers, and $\phi(x(0), x(T)) = \left[\begin{smallmatrix} x(0)-x_0 \\ x(T)-x_T \end{smallmatrix}\right]$. The second part forms a second-order cone complementarity problem (SOCCP) as follows:

$$-\eta \in \mathscr{K}, \quad Du \in \mathscr{K}, \quad \eta^T Du = 0, \tag{8}$$

where $\mathscr{K} = \mathscr{K}^d \times \mathscr{K}^d \times \cdots \times \mathscr{K}^d$.

From [19, 20], we see that the previous SOCCP (8) can be further recast as a system of equations:

$$\phi_{\mathrm{FB}}\left(Du, -\eta\right) = 0 \tag{9}$$

by employing the so-called complementarity function ϕ_{FB} which is a vector-valued function defined as

$$\phi_{\mathrm{FB}}(\mathbf{a}, \mathbf{b}) := \left(\mathbf{a}^2 + \mathbf{b}^2\right)^{1/2} - (\mathbf{a} + \mathbf{b}) \tag{10}$$

for $\mathbf{a} = \left[\begin{smallmatrix} a_1 \\ a_2 \end{smallmatrix}\right] \in \mathbb{R} \times \mathbb{R}^{d-1}, \mathbf{b} = \left[\begin{smallmatrix} b_1 \\ b_2 \end{smallmatrix}\right] \in \mathbb{R} \times \mathbb{R}^{d-1}$. We point out that the square term and square-root term in (10) are calculated via Jordan product

$$\mathbf{a} \circ \mathbf{b} = \begin{bmatrix} \mathbf{a}^{\mathbf{T}}\mathbf{b} \\ a_1 b_2 + b_1 a_2 \end{bmatrix}. \tag{11}$$

In particular, the expressions for $\mathbf{a}^2$ and $\mathbf{a}^{1/2}$ are given by

$$\mathbf{a}^2 = \begin{bmatrix} \|\mathbf{a}\|^2 \\ 2a_1 a_2 \end{bmatrix}, \tag{12}$$

$$\mathbf{a}^{1/2} = \begin{bmatrix} s \\ \dfrac{a_2}{2s} \end{bmatrix} \qquad \text{with } \mathbf{s} = \sqrt{\frac{1}{2}\left(\mathbf{a}_1 + \sqrt{\mathbf{a}_1^2 - \|\mathbf{a}_2\|^2}\right)}, \tag{13}$$

respectively.

In summary, the optimal grasping forces can be obtained by solving (7) and (9). Then, the semismooth Newton method is used to solve these equations, which will be described in next section.

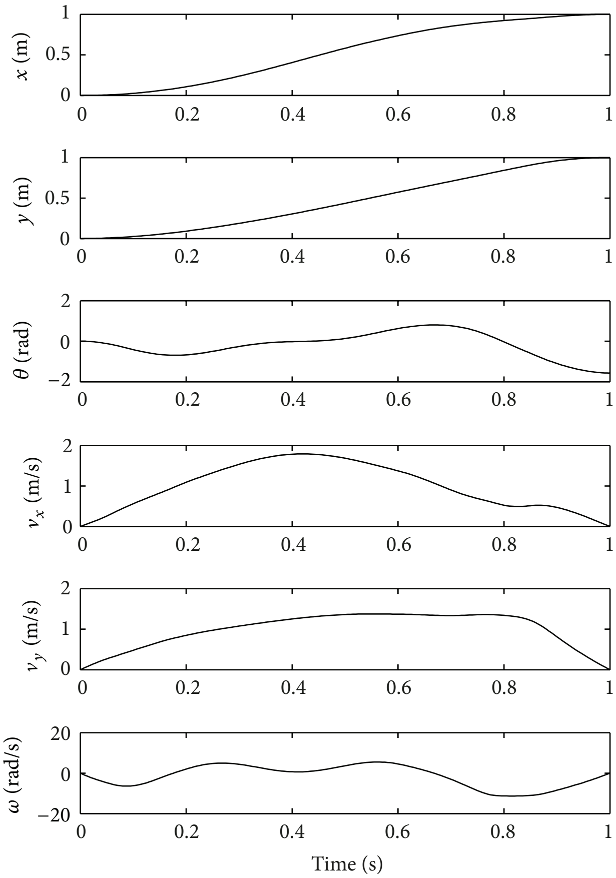

Figure 3: The trajectories of the variables x, y, θ, v_x, v_y, and ω in 90 degrees rotation simulation.

3. Semismooth Newton Method with Generalized Jacobian of FB Function

In order to apply the semismooth Newton method [21, 22] to (7) and (9), we need the following three linear equations:

$$\begin{bmatrix} \Delta\dot{x} \\ \Delta\dot{\lambda} \end{bmatrix} - \begin{bmatrix} f_x^{(k)} & 0 \\ -H_{xx}^{(k)} & -\left(f_x^{(k)}\right)^T \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta\lambda \end{bmatrix} - \begin{bmatrix} f_u^{(k)} & 0 \\ -H_{xu}^{(k)} & 0 \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta\eta \end{bmatrix} = -\begin{bmatrix} \dot{x}^{(k)} - f^{(k)} \\ \dot{\lambda}^{(k)} + H_x^{(k)} \end{bmatrix}, \tag{14}$$

$$\begin{bmatrix} \phi_{x(0)}^{(k)} & 0 & 0 \\ 0 & I & \left(\phi_{x(0)}^{(k)}\right)^T \\ 0 & 0 & -\left(\phi_{x(0)}^{(k)}\right)^T \end{bmatrix} \begin{bmatrix} \Delta x(0) \\ \Delta\lambda(0) \\ \Delta\sigma(0) \end{bmatrix} + \begin{bmatrix} \phi_{x(T)}^{(k)} & 0 & 0 \\ 0 & 0 & 0 \\ 0 & I & 0 \end{bmatrix} \begin{bmatrix} \Delta x(T) \\ \Delta\lambda(T) \\ \Delta\sigma(T) \end{bmatrix} = -\begin{bmatrix} \phi^{(k)} \\ \lambda^{(k)}(0) + \left(\phi_{x(0)}^{(k)}\right)^T \sigma^{(k)} \\ \lambda^{(k)}(T) + \left(\phi_{x(T)}^{(k)}\right)^T \sigma^{(k)} \end{bmatrix}, \tag{15}$$

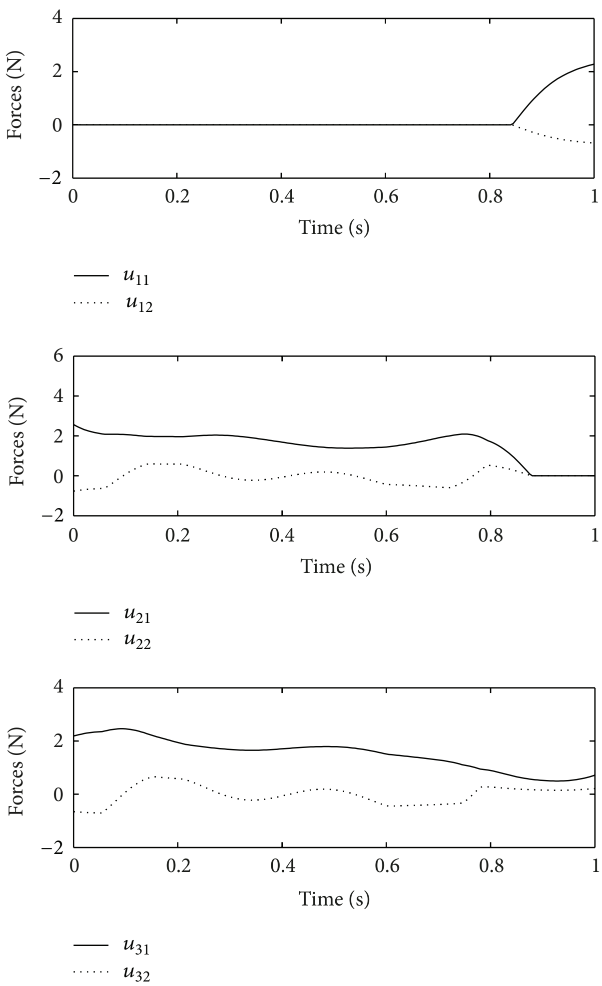

Figure 4: The trajectories of the grasping forces in 90 degrees rotation simulation.

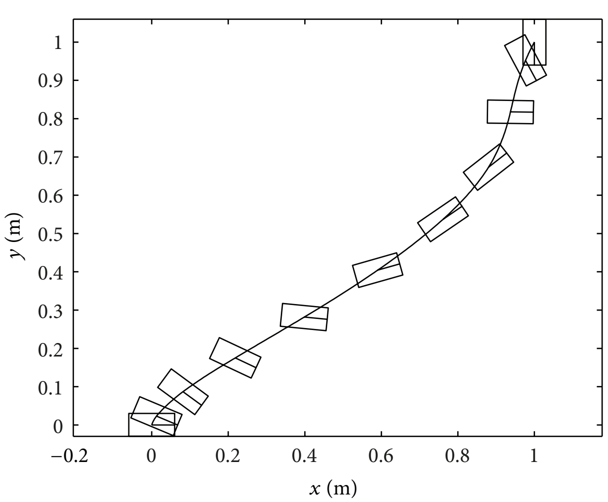

Figure 5: The manipulation path of 90 degrees rotation with $T = 1$ s and $\mu = 0.1$.

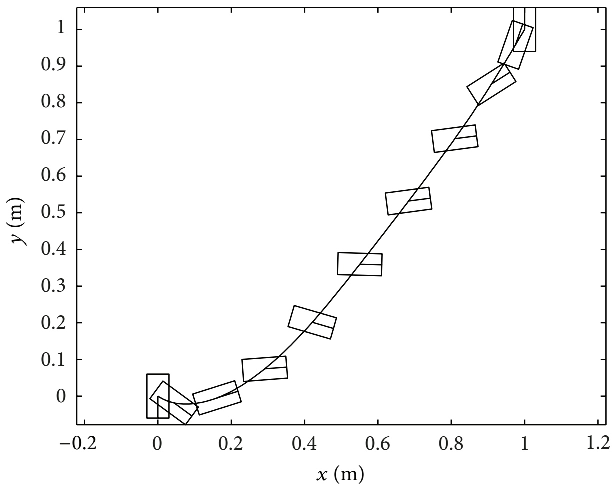

Figure 6: The manipulation path of 180 degrees rotation with $T = 1$ s and $\mu = 0.3$.

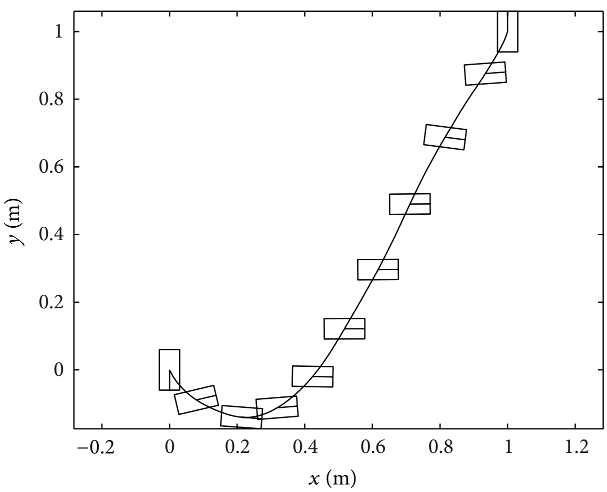

Figure 7: The manipulation path of 180 degrees rotation with $T = 2$ s and $\mu = 0.3$.

$$\begin{bmatrix} I & D^T \\ V_{\mathbf{a}}^{(k)} D & -V_{\mathbf{b}}^{(k)} \end{bmatrix} \begin{bmatrix} \Delta u \\ \Delta \eta \end{bmatrix} + \begin{bmatrix} \lambda^T f_{xu}^{(k)} & f_u^{(k)} \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \Delta x \\ \Delta \lambda \end{bmatrix} = - \begin{bmatrix} H_u^{(k)} \\ \phi_{\mathrm{FB}} \left(D u^{(k)}, -\eta^{(k)} \right) \end{bmatrix}. \tag{16}$$

Most information in linear equations (14)–(16) is known except the generalized Jacobian $V_{\mathbf{a}}$, $V_{\mathbf{b}}$ in (16). What do they represent? We provide a brief introduction here. First, we recall the concept of the B-subdifferential. Given a mapping $\Psi : \mathbb{R}^n \longrightarrow \mathbb{R}^m$, if Ψ is locally Lipschitz continuous, then the set

$$\partial_B H(z) := \left\{ V \in \mathbb{R}^{m \times n} \mid \exists \left\{ z^k \right\} \subseteq D_\Psi : z^k \longrightarrow z, \Psi' \left(z^k \right) \longrightarrow V \right\} \tag{17}$$

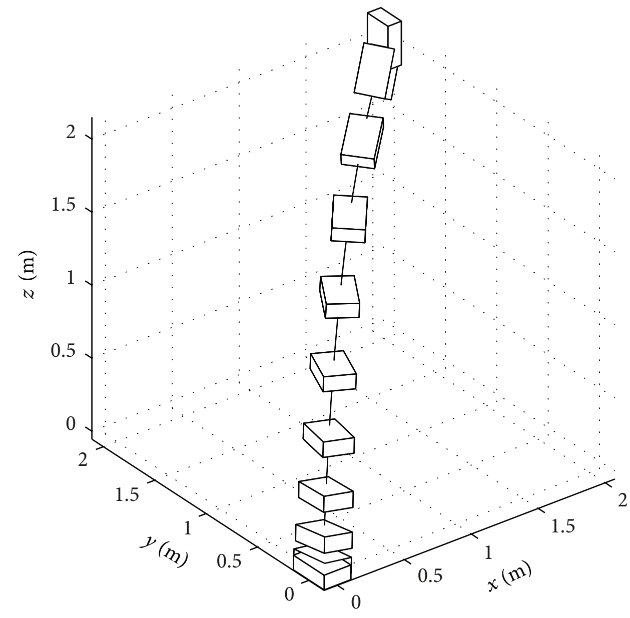

Figure 8: The 3D manipulation path of the five-fingered robot.

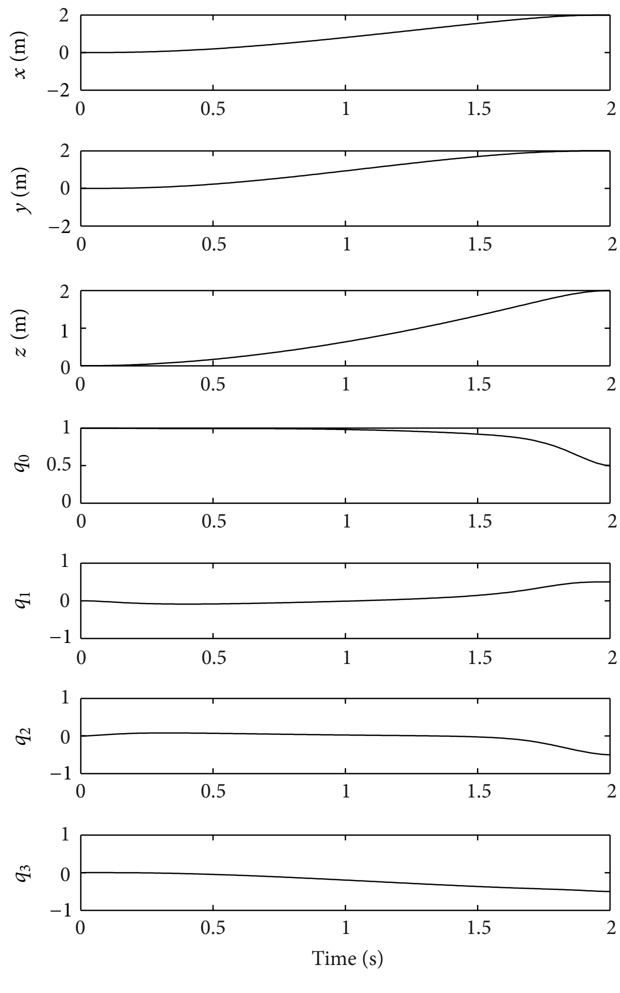

Figure 9: The trajectories of the variables $x, y, z, q_0, q1, q_2, q_3$ in five-fingered robot simulation.

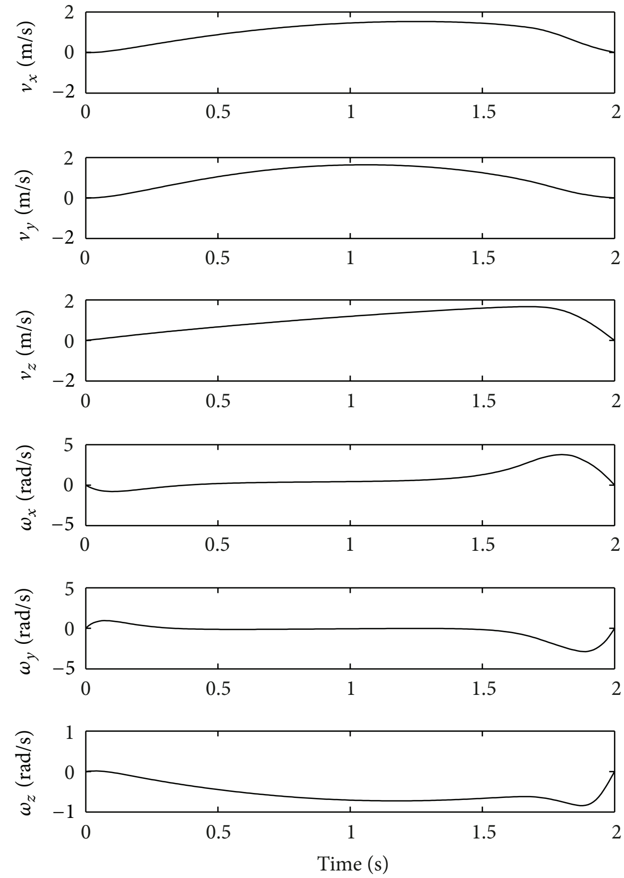

Figure 10: The trajectories of the variables $v_x, v_y, v_z, \omega_x, \omega_y, \omega_z$ in five-fingered robot simulation.

is nonempty and is called the B-subdifferential of Ψ at z, where $D_\Psi \subseteq \mathbb{R}^n$ denotes the set of points at which Ψ is differentiable. The convex hull

$$\partial\Psi(z) := \operatorname{conv}\partial_B\Psi(z) \tag{18}$$

is the generalized Jacobian of Clarke [23]. From this definition, we see that the generalized Jacobian of ϕ_{FB} can be obtained by computing $\partial_B\phi_{\mathrm{FB}}$. From [24, Proposition 3.1], the B-subdifferential of ϕ_{FB} in (16) is exactly expressed as

$$\partial_B\phi_{\mathrm{FB}}(\mathbf{a},\mathbf{b}) = \begin{bmatrix} V_{\mathbf{a}}^T \\ V_{\mathbf{b}}^T \end{bmatrix}. \tag{19}$$

Moreover, by denoting $\mathbf{c} := (\mathbf{a}^2 + \mathbf{b}^2)^{1/2}$, we have

(a) If $\mathbf{a}^2 + \mathbf{b}^2 \in \operatorname{int}(\mathscr{K}^{\mathbf{d}})$, then $V_{\mathbf{a}} = L_{\mathbf{c}}^{-1}L_{\mathbf{a}}$ and $V_{\mathbf{b}} = L_{\mathbf{c}}^{-1}L_{\mathbf{b}}$.

(b) If $\mathbf{a}^2 + \mathbf{b}^2 \in \operatorname{bd}(\mathscr{K}^{\mathbf{d}})$ and $(\mathbf{a},\mathbf{b}) \neq (\mathbf{0},\mathbf{0})$, then

$$V_{\mathbf{a}} \in \left\{ \frac{1}{2\sqrt{2w_1}} \begin{pmatrix} 1 & \overline{w}_2^T \\ \overline{w}_2 & 4I - 3\overline{w}_2\overline{w}_2^T \end{pmatrix} L_{\mathbf{a}} + \frac{1}{2}\begin{pmatrix} 1 \\ -\overline{w}_2 \end{pmatrix}\alpha^T \right\},$$

$$V_{\mathbf{b}} \in \left\{ \frac{1}{2\sqrt{2w_1}} \begin{pmatrix} 1 & \overline{w}_2^T \\ \overline{w}_2 & 4I - 3\overline{w}_2\overline{w}_2^T \end{pmatrix} L_{\mathbf{b}} + \frac{1}{2}\begin{pmatrix} 1 \\ -\overline{w}_2 \end{pmatrix}\beta^T \right\} \tag{20}$$

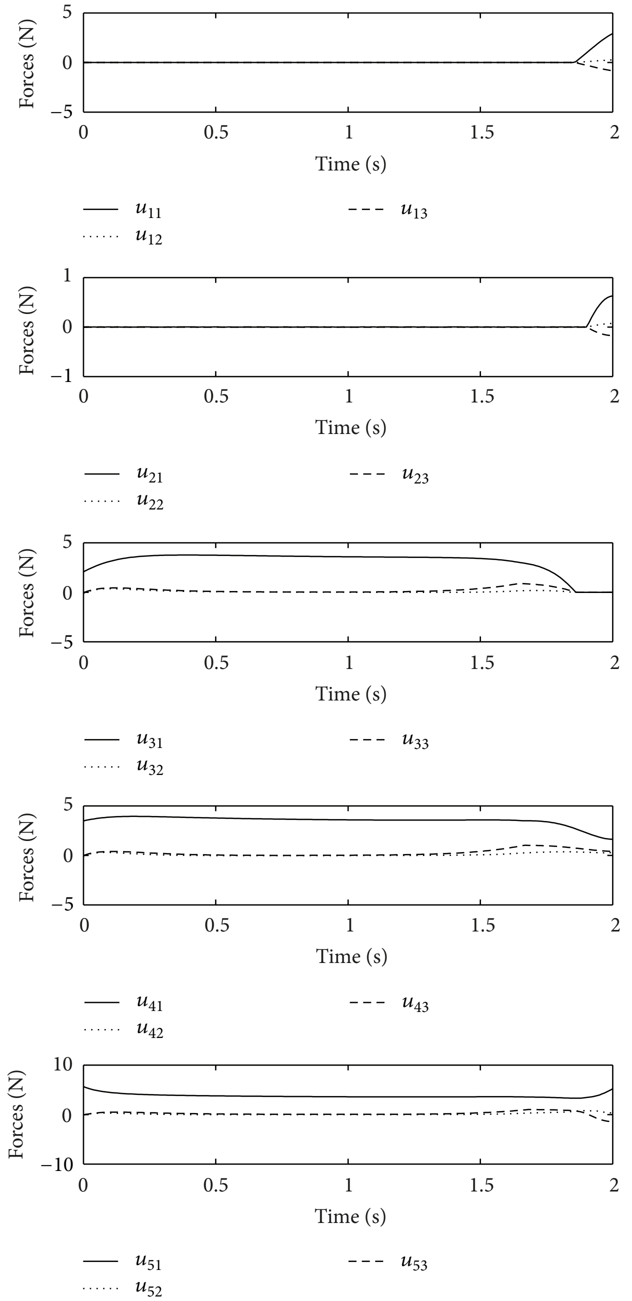

Figure 11: The trajectories of the grasping forces in five-fingered robot simulation.

for some $\alpha = \begin{bmatrix}\alpha_1\\ \alpha_2\end{bmatrix} \in \mathbb{R}\times\mathbb{R}^{d-1}$ $\beta = \begin{bmatrix}\beta_1\\ \beta_2\end{bmatrix} \in \mathbb{R}\times\mathbb{R}^{d-1}$ satisfying $|\alpha_1| \le \|\alpha_2\| \le 1$ and $|\beta_1| \le \|\beta_2\| \le 1$, where $\overline{w}_2 = w_2/\|w_2\|$.

(c) If $(\mathbf{a}, \mathbf{b}) = (\mathbf{0}, \mathbf{0})$, then $V_{\mathbf{a}} \in \{L_{\widehat{\mathbf{a}}}\}$, $V_{\mathbf{b}} \in \{L_{\widehat{\mathbf{b}}}\}$ for some $\widehat{\mathbf{a}}, \widehat{\mathbf{b}}$ with $\|\widehat{\mathbf{a}}\|^2 + \|\widehat{\mathbf{b}}\|^2 = 1$, or

$$V_{\mathbf{a}} \in \left\{\frac{1}{2}\begin{pmatrix}1\\ \overline{w}_2\end{pmatrix}\xi^T + \frac{1}{2}\begin{pmatrix}1\\ -\overline{w}_2\end{pmatrix}\alpha^T + 2\begin{pmatrix}0 & 0\\ \left(I - \overline{w}_2\overline{w}_2^T\right)\delta_2 & \left(I - \overline{w}_2\overline{w}_2^T\right)\delta_1\end{pmatrix}\right\},$$

$$V_{\mathbf{b}} \in \left\{\frac{1}{2}\begin{pmatrix}1\\ \overline{w}_2\end{pmatrix}\tau^T + \frac{1}{2}\begin{pmatrix}1\\ -\overline{w}_2\end{pmatrix}\beta^T + 2\begin{pmatrix}0 & 0\\ \left(I - \overline{w}_2\overline{w}_2^T\right)\gamma_2 & \left(I - \overline{w}_2\overline{w}_2^T\right)\gamma_1\end{pmatrix}\right\} \quad (21)$$

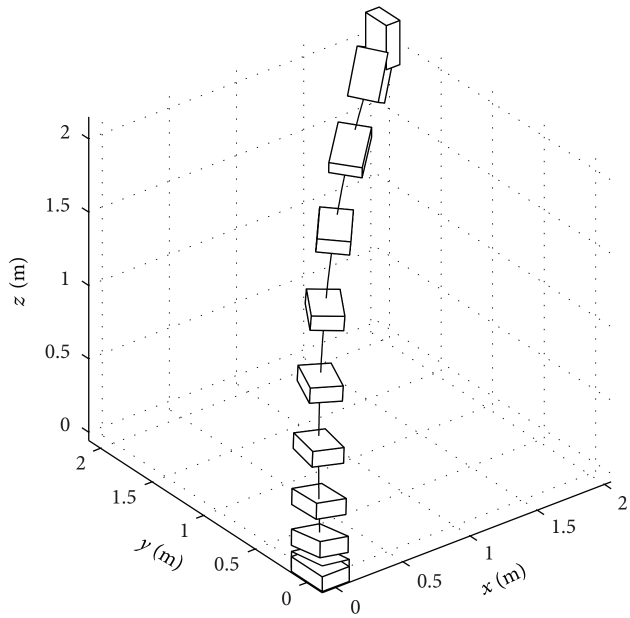

Figure 12: The 3D manipulation path of the four-fingered robot.

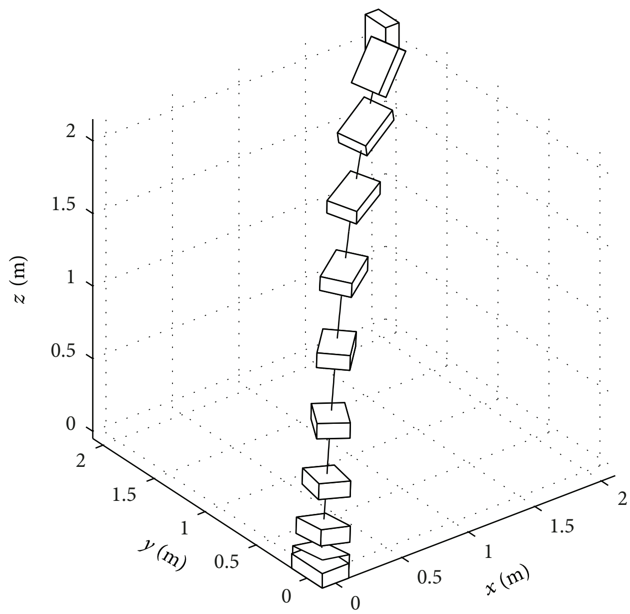

Figure 13: The 3D manipulation path of the six-fingered robot.

for some $\alpha = \begin{bmatrix}\alpha_1\\ \alpha_2\end{bmatrix}, \beta = \begin{bmatrix}\beta_1\\ \beta_2\end{bmatrix}, \xi = \begin{bmatrix}\xi_1\\ \xi_2\end{bmatrix}, \tau = \begin{bmatrix}\tau_1\\ \tau_2\end{bmatrix} \in \mathbb{R} \times \mathbb{R}^{l-1}$ such that

$$\begin{aligned} |\alpha_1| &\le \|\alpha_2\| \le 1, \qquad |\beta_1| \le \|\beta_2\| \le 1, \\ |\xi_1| &\le \|\xi_2\| \le 1, \qquad |\tau_1| \le \|\tau_2\| \le 1, \end{aligned} \tag{22}$$

$\overline{w}_2 \in \mathbb{R}^{l-1}$ satisfying $\|\overline{w}_2\| = 1$, and $\delta = \begin{bmatrix}\delta_1\\ \delta_2\end{bmatrix}, \gamma = \begin{bmatrix}\gamma_1\\ \gamma_2\end{bmatrix} \in \mathbb{R} \times \mathbb{R}^{d-1}$ satisfying $\|\delta\|^2 + \|\gamma\|^2 \le 1/2$.

Note that the calculations of $L_{\mathbf{a}}$ and $L_{\mathbf{a}}^{-1}$ are given by

$$\begin{aligned} L_{\mathbf{a}} &= \begin{bmatrix} a_1 & a_2^T \\ a_2 & a_1 I \end{bmatrix}, \\ L_{\mathbf{a}}^{-1} &= \frac{1}{a_1^2 - \|a_2\|^2} \begin{bmatrix} a_1 & -a_2^T \\ -a_2 & \dfrac{a_1^2 - \| a_2\|^2}{a_1} I + \dfrac{1}{a_1} a_2 a_2^T \end{bmatrix}. \end{aligned} \tag{23}$$

For more details, please refer to [24]. Now, we write down the iterative scheme of semismooth Newton method for solving the optimal grasping control problem.

Algorithm

Step 1. Choose $x^0, u^0, \lambda^0, \eta^0, \sigma^0$ and set $k = 0$.

Step 2. If convergence criterion is satisfied, stop.

Step 3. Compute the direction $\Delta x^k, \Delta u^k, \Delta \lambda^k, \Delta \eta^k, \Delta \sigma^k$ from the linear equations (14)–(16).

Step 4. Set

$$\begin{bmatrix} x^{k+1} \\ u^{k+1} \\ \lambda^{k+1} \\ \eta^{k+1} \\ \sigma^{k+1} \end{bmatrix} = \begin{bmatrix} x^{k} \\ u^{k} \\ \lambda^{k} \\ \eta^{k} \\ \sigma^{k} \end{bmatrix} + \begin{bmatrix} \Delta x^{k} \\ \Delta u^{k} \\ \Delta \lambda^{k} \\ \Delta \eta^{k} \\ \Delta \sigma^{k} \end{bmatrix}, \quad k = k + 1 \tag{24}$$

and go to Step 2.

A few words about the implementations. From (14) and (16), the parameters $\begin{bmatrix}\Delta u\\ \Delta \eta\end{bmatrix}$ can be eliminated and the differential equations regarding $\begin{bmatrix}\Delta x\\ \Delta \lambda\end{bmatrix}$ are obtained. With the boundary conditions (15), the solutions of $\begin{bmatrix}\Delta x\\ \Delta \lambda\end{bmatrix}$ can be achieved. Finally, the solutions of $\begin{bmatrix}\Delta u\\ \Delta \eta\end{bmatrix}$ can be obtained by (16). Once all linear equations are solved by the above procedures, the iterative scheme for the calculation of optimal grasping force is kept going.

4. Simulations

To evaluate the performance of the proposed approach, we do simulations for 2D and 3D multifingered robots for grasping manipulations. The 2D grasping simulations are performed with a plane three-fingered robot. The parameter values of the object are $m = 1\,\text{kg}$, $I = 0.01\,\text{kgm}^2$, $\mu = 0.6$, and the grasping matrices are

$$\begin{aligned} G_1 &= \begin{bmatrix} 0 & 1 & 0 & -1 & 0 & -1 \\ -1 & 0 & 1 & 0 & 1 & 0 \end{bmatrix}, \\ G_2 &= \begin{bmatrix} 0 & -0.03 & -0.03 & 0.03 & 0.03 & 0.03 \end{bmatrix}. \end{aligned} \tag{25}$$

The first 2D simulation is the manipulation of 90 degrees rotation of the object. The start and end points (x, y, θ) are set to be $(0\,\text{m}, 0\,\text{m}, 0\,\text{rad})$ and $(1\,\text{m}, 1\,\text{m}, -(\pi/2)\,\text{rad})$, respectively. Figure 2 shows the manipulation path of 90 degrees rotation with the time $T = 1\,\text{s}$ and the friction coefficient $\mu = 0.3$. The simulation result indicates that the proposed scheme grasps the object to the end point smoothly and accurately. Figure 3 depicts the trajectories of the variables x, y, θ, v_x, v_y, and ω. We observe that the rotation angle θ varies around 0 which results in a small grasping force. Moreover, the translation speeds are kept within 1.8 m/s and the turning speed within 5.5 rad/s. The trajectories of the grasping forces are shown in Figure 4. The simulation results show that the normal forces are all nonnegative and the tangent forces satisfy the friction constraint. To evaluate the effect of the friction, simulation with a different value of friction coefficient is also conducted for 90 degrees rotation simulation. Figure 5 shows the manipulation path with the friction coefficient $\mu = 0.1$. We observe that the manipulation path length becomes longer as the friction coefficient decreases. Meanwhile, the value of the objective function J is computed to be 3.07 when μ was 0.3 and it becomes 3.39 when μ reduces to 0.1, leading to the increase in grasping force.

The second 2D simulation is the manipulation of 180 degrees rotation of the object. The start and end points (x, y, θ) are set to be $(0\,\text{m}, 0\,\text{m}, -(\pi/2)\,\text{rad})$ and $(1\,\text{m}, 1\,\text{m}, (\pi/2)\,\text{rad})$, respectively. The friction coefficient is set as $\mu = 0.3$. Figures 6-7 depict the manipulation paths of 180 degrees rotation with the time $T = 1\,\text{s}$ and $T = 2\,\text{s}$, respectively. We observe that the object moves down initially and reaches to the end point accurately. Moreover, the mean of rotation angle decreases as T increases.

The 3D grasping simulation is performed with a five-fingered robot which has not been implemented in the literature. The object is considered as a block and its parameter values were set as

$$\begin{aligned} m &= 1\,\text{kg}, \qquad I_{11} = 8.33e - 3\,\text{kgm}^2, \\ I_{22} &= 4.17e - 3\,\text{kgm}^2, \qquad I_{33} = 1.08e - 2\,\text{kgm}^2. \end{aligned} \tag{26}$$

The two fingers of the robot grasp the top of the object, while the other three grasp the bottom of the object. The matrices G_1 and G_2 are

$$G_1 = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 \\ -1 & 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 & 1 & 0 & 0 \end{bmatrix},$$

$$G_2 = 0.01 \cdot \begin{bmatrix} 0 & 0 & 5 & 0 & 0 & 5 & 6 & 0 & 5 & -3 & 0 & 5 & -3 & 0 & 5 \\ 3 & 5 & 0 & -3 & 5 & 0 & 0 & -5 & 0 & -3\sqrt{3} & -5 & 0 & 3\sqrt{3} & -5 & 0 \\ 0 & 0 & -3 & 0 & 0 & 3 & 0 & -6 & 0 & 0 & 3 & 3\sqrt{3} & 0 & 3 & -5\sqrt{3} \end{bmatrix}. \quad (27)$$

The start and end points $(x, y, z, q_0, q_1, q_2, q_3)$ are set to be $(0\,\text{m}, 0\,\text{m}, 0\,\text{m}, 1, 0, 0, 0)$ and $(2\,\text{m},\ 2\,\text{m},\ 2\,\text{m},\ 0.5,\ \text{-}0.5,\ 0.5,\ \text{-}0.5)$, respectively. Figure 8 shows the 3D manipulation path with the time $T = 2\,\text{s}$ and the friction coefficient $\mu = 0.3$. As we can see, the object moves and rotates to the end point smoothly and accurately. Figures 9-10 depict the trajectories of the variables x, y, z, q_0, q_1, q_2, q_3 and v_x, v_y, v_z, ω_x, ω_y, ω_z, respectively. We observe that the trajectories are smooth. The translation speeds are kept within 1.67 m/s and the turning speed within 3.77 rad/s. The trajectories of the grasping forces are shown in Figure 11. The simulation results indicate that the grasping forces satisfy the friction constraint. The 3D grasping simulations are also performed with a four-fingered robot and a six-fingered robot, respectively. The four-fingered and six-fingered robots place one finger and three fingers on the top center of the object, respectively, while their other three fingers grasping the bottom. Figures 12-13 show the 3D manipulation paths with the four-fingered and six-fingered robots, respectively. From Figures 8, 12, and 13, we observe that increasing the number of the robot fingers can reduce manipulation path length and enhance the maneuverability. Consequently, the simulation results show that the proposed scheme can achieve the accurate manipulation for multifingered robots.

5. Conclusion

In this paper, we have proposed an effective method for multifingered robot path planning and grasping forces computation. The optimal grasping control problem was formulated with the rigid body dynamics of the object and the second-order cone constraints of grasping forces. The SOC complementarity problem was recast as the equations with the Fischer-Burmeister (FB) function, and the semismooth Newton method with the generalized Jacobian of FB function was used to solve the system equations. The simulation results show that the optimal grasping forces can accurately move the object to a goal, demonstrating the effectiveness of the proposed method.

Acknowledgments

This work was supported in part by the National Science Council of Taiwan, ROC under the Grant NSC 100-2221-E-214-016. The author's work is supported by National Science Council of Taiwan.

References

[1] A. Kawamura, K. Tahara, R. Kurazume, and T. Hasegawa, "Dynamic grasping for an arbitrary polyhedral object by a multi-fingered hand-arm system," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS '09)*, pp. 2264–2270, St. Louis, Mo., USA, October 2009.

[2] A. T. Miller and P. K. Allen, "Graspit: a versatile simulator for robotic grasping," *IEEE Robotics and Automation Magazine*, vol. 11, no. 4, pp. 110–122, 2004.

[3] R. M. Murray, Z. X. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*, CRC Press, Boca Raton, Fla, USA, 1994.

[4] T. Takahashi, T. Tsuboi, T. Kishida et al., "Adaptive grasping by multi fingered hand with tactile sensor based on robust force and position control," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '08)*, pp. 264–271, Pasadena, Calif, USA, May 2008.

[5] J. Xu and Z. Li, "A kinematic model of finger gaits by multifingered hand as hybrid automaton," *IEEE Transactions on Automation Science and Engineering*, vol. 5, no. 3, pp. 467–479, 2008.

[6] Y. Yokokohji, J. S. Martin, and M. Fujiwara, "Dynamic manipulability of multifingered grasping," *IEEE Transactions on Robotics*, vol. 25, no. 4, pp. 947–954, 2009.

[7] S. P. Boyd and B. Wegbreit, "Fast computation of optimal contact forces," *IEEE Transactions on Robotics*, vol. 23, no. 6, pp. 1117–1132, 2007.

[8] U. Helmke, K. Hüper, and J. B. Moore, "Quadratically convergent algorithms for optimal dextrous hand grasping," *IEEE Transactions on Robotics and Automation*, vol. 18, no. 2, pp. 138–146, 2002.

[9] L. Han, J. C. Trinkle, and Z. X. Li, "Grasp analysis as linear matrix inequality problems," *IEEE Transactions on Robotics and Automation*, vol. 16, no. 6, pp. 663–674, 2000.

[10] G. F. Liu, J. J. Xu, and Z. X. Li, "On geometric algorithms for real-time grasping force optimization," *IEEE Transactions on Control Systems Technology*, vol. 12, no. 6, pp. 843–859, 2004.

[11] Y. Zheng, M. C. Lin, and D. Manocha, "On computing reliable optimal grasping forces," *IEEE Transactions on Robotics*, vol. 28, no. 3, pp. 619–633, 2012.

[12] C. H. Ko, J. S. Chen, and C. Y. Yang, "Recurrent neural networks for solving second-order cone programs," *Neurocomputing*, vol. 74, no. 17, pp. 3646–3653, 2011.

[13] D. Prattichizzo, M. Malvezzi, M. Aggravi, and T. Wimbock, "Object motion-decoupled internal force control for a compliant multifingered hand," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA '12)*, pp. 1508–1513, St Paul, Minn, USA, May, 2012.

[14] F. L. Lewis and V. L. Syrmos, *Optimal Control*, John Wiley & Sons, New York, NY, USA, 1995.

[15] W. Ham and H. Kwon, "Sliding mode control for the hovering of helicopter by using quaternion dynamics," in *Proceedings of the 50th Annual Conference on Society of Instrument and Control Engineers (SICE '11)*, pp. 1024–1028, Tokyo, Japan, September 2011.

[16] C. G. Mayhew, R. G. Sanfelice, and A. R. Teel, "On quaternion-based attitude control and the unwinding phenomenon," in *Proceedings of the American Control Conference (ACC '09)*, pp. 299–304, San Francisco, Calif, USA, July 2011.

[17] H. P. Geering, *Optimal Control With Engineering Applications*, Springer, Berlin, Germany, 2007.

[18] D. G. Hull, *Optimal Control Theory for Applications*, Mechanical Engineering Series, Springer, Berlin, Germany, 2003.

[19] J.-S. Chen and P. Tseng, "An unconstrained smooth minimization reformulation of the second-order cone complementarity problem," *Mathematical Programming*, vol. 104, no. 2-3, pp. 293–327, 2005.

[20] M. Fukushima, Z. Q. Luo, and P. Tseng, "Smoothing functions for second-order-cone complementarity problems," *SIAM Journal on Optimization*, vol. 12, no. 2, pp. 436–460, 2002.

[21] M. Gerdts, "A nonsmooth Newton's method for control-state constrained optimal control problems," *Mathematics and Computers in Simulation*, vol. 79, no. 4, pp. 925–936, 2008.

[22] L. Q. Qi and J. Sun, "A nonsmooth version of Newton's method," *Mathematical Programming*, vol. 58, no. 3, pp. 353–367, 1993.

[23] F. H. Clarke, *Optimization and Nonsmooth Analysis*, vol. 5 of *Classics in Applied Mathematics*, SIAM, Philadelphia, Pa, 2nd edition, 1990.

[24] S. H. Pan and J.-S. Chen, "A damped Gauss-Newton method for the second-order cone complementarity problem," *Applied Mathematics and Optimization*, vol. 59, no. 3, pp. 293–318, 2009.