Dynamic Analysis and Control Synthesis of Grasping and Slippage of an Object Manipulated by a Robot

Shahram Hadian Jazi, Mehdi Keshmiri * and Farid Sheikholeslam
Isfahan University of Technology, Isfahan 8415683111, Iran
Received 26 February 2008; accepted 28 May 2008

Abstract
Grasping an object by a cooperating system such as multi-fingered hands and multi-manipulator robotic system has received much attention. Research has focused on analysis of force-closure grasps and the synthesis of optimal grasping, when there is no slipping condition. Although the control system is designed to keep the contact force in the friction cone and avoid the slipping condition, slippage can occur for many reasons. In this research, dynamics analysis and control synthesis of a manipulator moving an object on a horizontal surface using the contact force of an end-effector are performed considering the slipping condition. Equality and inequality equations of frictional contact conditions are replaced by a single second-order differential equation with switching coefficients in order to facilitate the dynamic modeling. Accuracy of this modeling is verified by comparing the results of the model with those of SimMech. Using this modeling of friction, a set of reduced order form is obtained for equations of motion of the system. A new method is proposed to control the object motion and the end-effector undesired slippage based on the reduced form. Finally, performance of the method is evaluated both numerically and experimentally.


Keywords
Grasping, contact modeling, slippage control, frictional contact, undesired slipping, cooperating system

1. Introduction
Grasping an object in a cooperating system such as multi-fingered hands and multiple robots is an important issue for researchers. During a full multi-fingered manipulation cycle, grasp planning arises on several occasions, such as when an object is first picked up. Grasp analysis and synthesis are fundamental problems in the study of grasp planning. Many papers can be found on testing and planning force-closure grasps. In grasp analysis, most of the research has been focused on finding appropriate conditions for force-closure grasps. Previously, Salisbury and Roth developed
several different types of finger contacts and showed which finger configurations allow complete immobilization of the gripped object relative to the fingers, and also allow for the manipulation of the object by the fingers while maintaining the grasp, using screw theory [1]. Mishra et al. proposed necessary and sufficient conditions for force-closure grasp with friction point contacts (FPCs) [2]. Bicchi translated the force-closure problem into the stability of an ordinary differential equation [3]. With the linearization of the friction cone, Liu developed a ray-shooting-based algorithm using the duality of polytopes [4]. Zheng and Qian enhanced the ray-shooting approach proposed by Liu to complete the exactness, increase the efficiency and extend the scope [5]. Using linear matrix inequality representation of nonlinear friction cone constraints, Han et al. reformulated the force-closure problem as the feasibility problem of a semi-definite or max-det problem and used an interior point algorithm for it [6]. Thus, the general problem of determining if a grasp is a force-closure grasp is considered to be completely solved.

Having sufficient conditions for force closure, grasp synthesis deals with optimal grasping. This synthesis consists of (i) determination of the optimality criteria, and (ii) derivation of methods and algorithms for computing contact locations with respect to the optimality criteria and subject to accessibility constraints. Park and Starr presented a simple and efficient algorithm to find an optimum force-closure grasp of a planar polygon using a three-fingered robot hand. The optimum grasp was defined as a grasp that has the minimum value of a heuristic function [7]. Mishra et al. proposed an algorithm for computing force-closure grasps for polyhedral objects under FPCs [2]. Tung and Kak presented a new theorem and an algorithm for the fast synthesis of two-fingered force-closure grasps for arbitrary polygonal objects. The polygonal objects were allowed to be of arbitrary shape and each edge of the polygon was allowed to have different frictional characteristics [8]. Liu presented an efficient algorithm for computing all n-finger form-closure grasps on a polygonal object based on a new sufficient and necessary condition for form-closure. With this new condition, it is possible to transfer the problem of computing the form-closure grasp in $\mathbb{R}^3$ to one in $\mathbb{R}^1$ [9]. Based on the geometric condition of the closure property, Zhu and Ding presented a numerical test to quantify how far a grasp is from losing form/force closure. They also developed an iterative algorithm for computing optimal force-closure grasps [10]. Morales et al. addressed the problem of designing a practical system able to grasp real objects with a three-fingered robot hand. They presented a general approach for synthesizing two- and three-finger grasps on planar unknown objects using visual perception [11]. Al-Gallaf presented a novel neural network for dexterous hand-grasping inverse kinematics mapping used in force optimization. He showed that the proposed optimization is globally convergent to the optimal grasping force [12]. Zhu et al. introduced the $Q$ distance and adopted the radius of the largest ball contained in the convex hull of the primitive wrenches as a quality measure [13, 14]. Platt et al. presented an algorithm for grasping an unknown object by cooperating robots. They found proper contact points such that resultant couple and force on the object are zero, and controllability of nor-
mal force control or tangential velocity control was maximized [15]. Miyabe et al. analyzed object grasping with elastic arms. They used optimized contact velocity to find forces that can grasp object [16]. Li et al. presented a novel algorithm for computing three-finger force-closure grasps of two-dimensional (2-D) and three-dimensional (3-D) objects. New necessary and sufficient conditions for 2-D and 3-D equilibrium and force-closure grasps were deduced, and a corresponding algorithm for computing force-closure grasps was developed [17]. Liu et al. proposed a complete and efficient algorithm for searching form-closure grasps of $n$ hard fingers on the surface of a 3-D object represented by discrete points [18]. All of the above research considers no slippage in grasping and the control system tries to keep control forces within in the friction cone.

Zheng et al. addressed dynamic and control analysis of a three-fingered hand manipulating and regrasping an object in 3-D space. They allowed one of the fingers to slide on a predefined path on the object surface to change its grasp location [19]. Cole et al. consider control of the sliding motion of the fingertip of a two-fingered hand along the object surface and position and orientation control of the object simultaneously. They assumed that only one specific finger slides on a predefined path on the object surface. Their work is useful for regrasping an object held in a hand [20]. Kao and Cutkosky compared theoretical and experimental sliding motions for a sheet of paper or similar objects on a planar surface, manipulated by a two-fingered hand, using static equilibrium equations [21]. Chong et al. proposed a motion/force planning algorithm for multi-fingered hands manipulating an object of an arbitrary shape using both rolling and sliding contacts. They used a nonlinear optimization approach to calculate the joint velocities and contact forces at each step of time [22].

Although the above research considers slippage in object regrasping analysis, the slippage should be completely predefined. The finger which slides on the object, starting time and duration of slippage, and sliding path are known in advance. This means dynamic and control analysis of undesired slippage still remains undiscussed in the literature. It can occur during the grasping maneuver due to many reasons, such as changes in the object geometry, mass, inertia and coefficient of friction or dealing with an unknown object. As an example, one can assume the practical case when a cooperative system manipulates a dirty object or manipulates an object in a dirty circumstance. In such a case, the coefficient of friction between end-effectors and object can change.

In this research, dynamic analysis and control synthesis of a manipulator moving an object on a surface using contact forces and considering undesired slipping conditions have been performed. The system under consideration is a platform for further extension to grasping analysis of a cooperative manipulator, carrying an object while slipping conditions can occur. In Section 2, problem definition and assumptions are given, and general formulation and equations of motion for the system are provided. In Section 3 dynamics of friction are formulated. In order to control sliding of the object, as well as its path tracking, two types of controllers
have been proposed in Sections 4 and 5. Section 6 provides the numerical results. An experimental setup built for implementing the method practically is described in Section 7. Comparison of the experiment results with those of the numerical simulation are also provided in this section.

1.1. Dynamic Analysis

The system under consideration is shown in Fig. 1. It consists of a two-link rigid manipulator that moves an object (B) on the horizontal surface. Contact between the manipulator and object is assumed to be point contact which can be moved along the object surface.

The whole motion is assumed to be in the vertical plane and the following assumptions are taken into account as well:

- The contact point on the end-effector remains fixed. Note that without this assumption, an additional kinematics problem must be considered.
- The object cannot rotate or its inertia momentum is zero.
- The coefficient of friction on the top of the object is greater than the bottom.
- There is no uncertainty in the system.

Using the above assumptions, the motion of the whole system can be described by the following equations:

\[ \dot{M}\ddot{q} + h = B\tau - J^TF_1 \]  \hspace{1cm} (1)

\[ M_0\ddot{x}_b + g_0 = G\dot{F} \]  \hspace{1cm} (2)

\[ H_u(F_1, \ddot{x}_s) = 0 \]  \hspace{1cm} (3)

\[ H_d(F_2, \ddot{x}_b) = 0 \]  \hspace{1cm} (4)

where \( M, h, B \) and \( J \) are the well-known matrices and vectors of the two-link manipulator and:

![Figure 1. Schematic of the system under consideration.](image)
\[ M_0 = \begin{bmatrix} m_b \\ 0 \end{bmatrix} \]
\[ g_0 = \begin{bmatrix} 0 \\ m_b g \end{bmatrix} \]
\[ G = [I_2 \ I_2] \]
\[ F = \begin{bmatrix} F_1 \\ F_2 \end{bmatrix}. \]  

\( F_1 \) and \( F_2 \) are the forces exerted on the object by end-effector and ground, respectively, \( x_b \) and \( x_s \) are the movements of the object on the horizontal surface and end-effector on the object, respectively, and \( m_b \) is the mass of the manipulated object. \( H_u \) and \( H_d \) are functions which model friction on the surfaces of the object. They are described in the next section.

System motion is a kind of constrained motion and the constraint equations can be written as

\[ x_e = x_b + x_s \]
\[ y_e = \text{const.}, \]  

where end-effector position, \( x_e \) and \( y_e \), can be written as:

\[ x_e = l_1 \cos(q_1) + l_2 \cos(q_1 + q_2) \]
\[ y_e = l_1 \sin(q_1) + l_2 \sin(q_1 + q_2). \]  

Substituting for \( x_e \) and \( y_e \) from (7) in (6) and differentiating with respect to time, the constraint equations can be written as the following velocity form:

\[ A_q \ddot{q} + A_x \dot{x}_b + A_s \dot{x}_s = 0, \]  

where:

\[ A_q = J, \quad A_x = A_s = \begin{bmatrix} -1 \\ 0 \end{bmatrix}. \]  

2. Modeling the Contact Forces

Friction appears at the physical interface between two surfaces when two bodies are in contact, which is strongly influenced by contamination. There is a wide range of physical phenomena that cause friction. These include elastic and plastic deformations, fluid mechanics and wave phenomena, and material sciences (see Refs [23, 24]). A survey of models of friction analysis can be found in Ref. [25].
Assuming the Standard Coulomb friction model without stiction (Fig. 2) with \( \mu \) as the coefficient of friction \((\mu_s = \mu_k = \mu)\), the friction force exerted on a body from the contacting surface (Fig. 3) can be written as:

\[
\begin{align*}
F_t &= -\mu F_n \text{sign}(v) & \text{if } v \neq 0 \\
|F_t| &\leq \mu F_n & \text{if } v = 0 \text{ and } \dot{v} = 0 \\
F_t &= 0 & \text{if } v = 0 \text{ and } \dot{v} \neq 0,
\end{align*}
\]  

(10)

where \( v \) is the speed of the body relative to the surface, and \( F_t \) and \( F_n \) are friction and normal forces, respectively. \( F_n \) is assumed to be a positive value.

Note that the second equation in (10) describes three different conditions: starting forward motion, starting backward motion and stationary condition. We can reformulate the above conditions in a single equation:

\[
\alpha_1 \dot{v} + \alpha_2 F_t + \alpha_3 \mu F_n = 0,
\]

(11)

where \( \alpha_i \) \((i = 1, 2, 3)\) are state-dependent coefficients calculated from Table 1.

When there is more than one choice for \( \alpha_i \) \((i = 1, 2, 3)\), we have to choose one set and check the selection for consistency with the result from dynamic analysis.

For the simulation and control purposes, detecting the zero velocity is a problem due to numerical issues. A remedy for this can be found in the model presented by Karnopp [26]. It was developed to overcome the problem and to avoid switching between different state equations for sticking and sliding. The model defines a zero-velocity interval, \(|v| < \delta v\). For velocities within this interval the internal state
Table 1. Values for $\alpha_i$ ($i = 1, 2, 3$) in different conditions

<table>
<thead>
<tr>
<th>$\alpha_i$</th>
<th>$v = 0$</th>
<th>$\dot{v}^- = 0$</th>
<th>$\dot{v}^- = 0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Movement</td>
<td>No Motion</td>
<td>No Motion</td>
<td>Start forward</td>
</tr>
<tr>
<td>$\alpha_1$</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>$\alpha_2$</td>
<td>1</td>
<td>1</td>
<td>0</td>
</tr>
<tr>
<td>$\alpha_3$</td>
<td>sign($v$)</td>
<td>0</td>
<td>0</td>
</tr>
</tbody>
</table>

Figure 4. Free body diagram for the object in Fig. 1.

of the system (the velocity) may change and be non-zero, but the output of the block is maintained at zero by a dead zone. The drawback with this model is that it is so strongly coupled with the rest of the system and it is not always explicitly given. In fact this is the reason that we have to choose one option and check for its consistency. This model is utilized in the presented research.

Let us consider the object shown in Fig. 1. Its free body diagram is given in Fig. 4. The contact conditions can be formulated by:

$$H_u(F_1, \ddot{x}_s) = \alpha_1 \ddot{x}_s + \alpha_2 F_{1x} + \alpha_3 \mu_1 F_{1y} = 0$$
$$H_d(F_2, \ddot{x}_b) = \beta_1 \ddot{x}_b + \beta_2 F_{2x} + \beta_3 \mu_2 F_{2y} = 0,$$

where $\alpha_i$ and $\beta_i$ ($i = 1, 2, 3$) are calculated from Table A.1 in Appendix A.

The results of the above modeling are compared with those of the SimMech toolbox of MATLAB software which uses differential-algebraic equations. Surprisingly, we obtained the same results.

3. Control Synthesis — Conventional Approach

In the conventional approach of grasping analysis the controller is designed such that the manipulator exerts the requested force on the object and satisfies the no slipping condition. It is worth mentioning that, in most of the existing research,
researchers solve the problem for the case that there is no slippage. Although we have used the same approach in this section, we extend the method and previous works for the case that the end-effector slips on the object.

Consider equations of motion for the system (2) and let the desired trajectory of the object be \( x_b^{\text{des}}(t) \). If the acceleration of the object is chosen as:

\[
\ddot{x}_b = \ddot{x}_b^{\text{des}} + K_{vb} \dot{e}_b + K_{pb} e_b,
\]

where \( K_{vb} \) and \( K_{pb} \) are positive constants and \( e_b = x_b^{\text{des}} - x_b \), and the following resultant force is applied on the object:

\[
F_{\text{res}} = GF = M_0 (\ddot{x}_b^{\text{des}} + K_{vb} \dot{e}_b + K_{pb} e_b) + g_0.
\]

then the object’s motion is governed by:

\[
\ddot{e}_b + K_{vb} \dot{e}_b + K_{pb} e_b = 0.
\]

This guarantees asymptotic stability of the tracking.

Now one has to decompose the resultant force into the exerted forces on the object by the manipulator and horizontal surface, and then control the robot in such a way to ensure that the calculated forces for the manipulator are implemented.

Due to the redundancy in the driving forces of the object, decomposition of the resultant force leads to the following optimization problem:

Minimize \( \|F^{\text{des}}\| \)

such that:

\[
F_{\text{res}} = GF^{\text{des}}
\]

\[
e_M^T f_1^{\text{des}} \geq \eta_1 \|F_1^{\text{des}}\|
\]

\[
e_M^T f_2^{\text{des}} = \eta_2 \|F_2^{\text{des}}\|
\]

\[
e_M^T f_1^{\text{des}} > 0
\]

\[
e_M^T f_2^{\text{des}} > 0,
\]

where \( \eta_i = \frac{1}{\sqrt{1+\mu_i^2}} \) and \( e_N^i \) is inward normal direction in the \( i \)th contact point.

In (17) we have used \( F^{\text{des}} \) instead of \( F \) since the exerted forces by the manipulator and the horizontal surface can differ from the calculated ones. Contact stability can be deteriorated once the manipulator cannot exert the desired forces. In this case the end-effector will slip on the object.

Since the end-effector forces must be controlled in all directions, the usual hybrid position/force control cannot be used. Thus, we design the controller of the manipulator such that the desired forces are exerted by the end-effector and no slippage condition is satisfied in the contact point, i.e.:

\[
\dot{x}_e = \ddot{x}_b.
\]
If we divide the input torques in (1) into two parts, $\tau = \tau_e + \tau_f$, where $\tau_e$ and $\tau_f$ are responsible for satisfying the no slippage condition (18), and exerting the calculated force, $F_{\text{des}}^1$ on the object, respectively, we can compute $\tau_f$ from the static equilibrium condition:

$$B\tau_f = J^TF_{\text{des}}^1,$$

(19)

and $\tau_e$ from the free motion of manipulator ($M\ddot{q} + h = B\tau_e$). In this research a feedback linearization method is used to control the free motion of the manipulator.

One must note that using this approach cannot always generate any arbitrary pair of $F_1$ and $\dot{x}_e$ [27]. However, since $\dot{x}_e$ is somehow the result of $F_1$, the above approach can result in the desired objectives. The fundamental structure of this controller is shown in Fig. 5.

Two different strategies can be imagined to control slipping of the end-effector on the object. In the first strategy we set the velocity of the object as the desired velocity for the end-effector and the contact point position for its desired position, i.e.:

$$x_{\text{des}}^e = x_b + x_s$$

$$\dot{x}_{\text{des}}^e = \dot{x}_b.$$  

(20)

This means in each instant we try to stop slipping, without trying to return the contact point to its original position. In the second strategy the end-effector always tries to return to its original position on the object, i.e.:

$$x_{\text{des}}^e = x_b.$$  

(21)

In Appendix B, it is shown that the first strategy always results in a steady-state error in velocity, while the second strategy will conclude to zero error as soon as the desired force objective is met by the controller. This advantage is shown in the numerical analysis as well.

![Figure 5. System description in the conventional approach.](image)

As can be seen, the conventional approach is an open-loop controller in force control and a closed-loop controller in position control. Since avoiding slippage in this approach depends on the error in the exerted force, one can expect fluctuation around the zero-slipping condition. Also we cannot perform stability analysis for this method.

Now a new approach in designing the controller is proposed. In this approach we convert the constrained equations of motion into two inputs, two outputs, a non-constrained motion and design the controller for the new sets of equations.

Let us rewrite (12) and (13) as:

\begin{align}
\alpha_1 \ddot{x}_s + D_1 F_1 &= 0 \quad (22) \\
\beta_1 \ddot{x}_b + D_2 F_2 &= 0, \quad (23)
\end{align}

where:

\[
D_1 = [\alpha_2 \quad \alpha_3 \mu_1 ]
\]

\[
D_2 = [\beta_2 \quad \beta_3 \mu_2 ].
\]

Calculating \(F_1, F_2\) and \(\ddot{q}\) from (1), (2) and the derivative of (8), respectively, and substituting them in (22) and (23), one can write the following reduced order form:

\[
\tilde{M} \begin{bmatrix} \ddot{x}_b \\ \ddot{x}_s \end{bmatrix} + \tilde{h} = \tilde{B} \tau,
\]

where:

\[
\tilde{M} = \begin{bmatrix}
D_1 J^{-T} M A_q^{-1} A_x & \alpha_1 + D_1 J^{-T} M A_q^{-1} A_s \\
\beta_1 + D_2 M_0 - D_2 J^{-T} M A_q^{-1} A_x & -D_2 J^{-T} M A_q^{-1} A_s
\end{bmatrix}
\]

\[
\tilde{h} = \begin{bmatrix}
-D_1 J^{-T} M A_q^{-1} b - D_1 J^{-T} h \\
D_2 g_0 + D_2 J^{-T} M A_q^{-1} b + D_2 J^{-T} h
\end{bmatrix}
\]

\[
\tilde{B} = \begin{bmatrix}
-D_1 \\
D_2
\end{bmatrix} J^{-T} B
\]

\[
b = -(\dot{A}_q \dot{q} + \dot{A}_s \dot{x}_b + \dot{A}_s \dot{x}_s).
\]

Note that all of the above matrices and vectors depend on \(q, \dot{x}_b\) and \(\dot{x}_s\), and their derivatives.

Equation (25) relates input and output of the system. In order to use this equation to design the controller, one has to take care of the internal stability of the system and to show that \(\tilde{M}\) is invertible.

**Theorem 1.** The system of Fig. 1 represented by (25) is internally stable, if it is input–output stable.
\textbf{Proof.} Assuming input–output stability means that $\tau$, $x_b$, and $x_s$ are bounded. Since $x$-position of the end-effector can be written as:

$$x_e = x_b + x_s,$$

(28)

it means $x_e$ and its derivatives are also bounded. Considering this fact and the kinematics relation between $\dot{q}$ and end-effector velocity, and noting that $q$ appears in trigonometric equations, results that $\dot{q}$ is bounded. Using (1) and (2), one can show that $F_1$ and $F_2$ are also bounded.

\textbf{Theorem 2.} $\bar{M}$ is invertible if and only if:

$$\beta_1 I_2 \leq \int_{t_0}^{t_0+T} \bar{M}_d \bar{M}_d^T dt \leq \beta_2 I_2,$$

(29)

for all $t_0$, where $\bar{M}_d = \bar{M}(x_b^{\text{des}}, x_s^{\text{des}})$. $\beta_1$, $\beta_2$ and $T$ are positive constant scalars. \textit{Proof is given in Appendix C.}

\textbf{Remark.} Since the determinant is a continuous function of the matrix elements, for a matrix whose elements are a continuous function of a parameter such as time the determinant will also be a continuous function of time. Thus, in any interval that determinant is non-zero the matrix is invertible.

Since we are facing a multi-phase dynamic system, we propose an appropriate multi-phase controller such that it complies with the system.

As far as slipping is concerned, the system undergoes the following four phases:

(1) Slipping on both sides.

(2) Slipping only on the bottom.

(3) Slipping only on the top.

(4) No slipping on either side.

Each of these four phases covers part of the $\dot{x}_b$–$\dot{x}_s$ plane. As shown in Fig. 6, phase 1 covers the whole plane excluding the axes. Phases 2 and 3 cover $\dot{x}_b$ and $\dot{x}_s$ axes, respectively, excluding the origin. Phase 4 covers only the origin.

When the system is in phase 1, the controller must prevent slipping in the upper side of the object and keep the object moving such that it tracks the desired trajectory. In phase 2, there is no slipping in the upper side of the object and the controller must only ensure that the object tracks the desired trajectory. Phase 3 and 4 are in fact non-desirable phases, and the controller must enforce the system to leave these phases and enter in phase 1 or 2.

Using this strategy, the controller pushes the system working always in phase 1 or 2, except for the limited period of time where the object stops and the controller tries to move it. Hence, any stable and convergent control law for phase 1 and 2 guarantees stability and convergence of the whole motion. We can guarantee that in
phase 3 and 4 the controller puts the object in motion in a limited time and different force condition from the time that the object stops.

The following control law has been used for phase 1 and 2, for the sake of numerical simulation:

$$\tilde{M}(\ddot{X}^{\text{des}} + K_v \dot{e} + K_p e) + \tilde{h} = \tilde{B} \tau,$$

where:

$$\ddot{X}^{\text{des}} = \begin{bmatrix} \ddot{x}_b^{\text{des}} \\ 0 \end{bmatrix},$$

$$e = \begin{bmatrix} e_b \\ e_s \end{bmatrix} = \begin{bmatrix} x_b^{\text{des}} - x_b \\ 0 - x_s \end{bmatrix}.$$ (32)

$K_p$ and $K_v$ are two positive definite matrices that regulate frequency and speed of tracking convergence.

Therefore, input vector, $\tau$, can be calculated as:

$$\tau = \tilde{B}^+ \tilde{M}(\ddot{X}^{\text{des}} + K_v \dot{e} + K_p e) + \tilde{B}^+ \tilde{h} + (I - \tilde{B}^+ \tilde{B})y,$$ (33)

where $\tilde{B}^+$ is the Moore–Penrose pseudoinverse of $\tilde{B}$ and $y$ is an arbitrary vector.

Substituting (33) in (1) and eliminating $F_1$ and $F_2$ and $\ddot{q}$ from (1), (2) and the derivative of (8), the following equation is obtained:

$$\tilde{M} \ddot{X} + \tilde{h} - \tilde{B} \tilde{B}^+ \tilde{M}(\ddot{X}^{\text{des}} + K_v \dot{e} + K_p e) - \tilde{B} \tilde{B}^+ \tilde{h} - \tilde{B}(I - \tilde{B}^+ \tilde{B})y = 0.$$ (34)

Since $\tilde{B}(I - \tilde{B}^+ \tilde{B}) = 0$, (34) can be rewritten as:

$$\tilde{M} \ddot{X} + \tilde{h} - \tilde{B} \tilde{B}^+ \tilde{M}(\ddot{X}^{\text{des}} + K_v \dot{e} + K_p e) - \tilde{BB}^+ \tilde{h} = 0,$$ (35)

defining:

$$P = \tilde{BB}^+,$$ (36)
and performing some matrix operation, one can rewrite (35) as:

\[(I - P)(\ddot{M} \dot{X} + \dot{h}) - PML(e) = 0,\]  

(37)

where:

\[L(e) = \ddot{e} + K_v \dot{e} + K_p e.\]  

(38)

In phase 1 \(D_1 \neq 0\) and \(D_2 \neq 0\). Hence, \(\tilde{B}\) is invertible and \(\tilde{B}^+ = \tilde{B}^{-1}\), so:

\[P = I,\]  

(39)

and then:

\[\tilde{M}L(e) = 0.\]  

(40)

\(L(e)\) becomes continuously zero and it guarantees the error convergence, since the matrix \(\tilde{M}\) is full rank.

Physical interpretation: the proposed controller controls slippage on the upper side and reduces the object motion error by increasing the normal force exerted on the object by the end-effector. This causes a change in the tangential forces, \(F_{1x}\) and \(F_{2x}\), such that the object tracks the desired trajectory and the end-effector slippage on the object diminishes.

In phase 2, the first row of \(\tilde{B}\) becomes zero. It can be shown that in this phase:

\[P = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix},\]  

(41)

and:

\[(I - P)(\ddot{M} \dot{X} + \dot{h}) = \begin{bmatrix} \ddot{x}_s \\ 0 \end{bmatrix}.\]  

(42)

Since there is no sliding in the upper face, \(\ddot{x}_s = 0\) and we can write:

\[(I - P)(\ddot{M} \dot{X} + \dot{h}) = 0.\]  

(43)

Using the above relation in (37) leads to:

\[PML(e) = \begin{bmatrix} 0 \\ m_{21}(\ddot{e}_b + k_{vb} \dot{e}_b + k_{pb} e_b) + m_{22}(\ddot{e}_s + k_{vs} \dot{e}_s + k_{ps} e_s) \end{bmatrix} = 0.\]  

(44)

Note that in this phase \(\ddot{x}_s = x_{s}^{\text{des}} = 0\) and \(\ddot{x}_s = x_{s}^{\text{des}} = 0\). So we will have:

\[m_{21}(\ddot{e}_b + k_{vb} \dot{e}_b + k_{pb} e_b) + m_{22}k_{ps} e_s = 0.\]  

(45)

Selecting \(k_{ps} = 0\) in this phase guarantees the tracking error of the object converges to zero.

Since the dimension of null space of \(\tilde{B}\) is not zero, \(y\) can have infinite choices. It allows us to compute \(\tau_1\) and \(\tau_2\) such that the norm of contact forces becomes
minimum. Hence, for this case we solve the following problem:

Minimize \( \| F(y) \| \)

such that

\[
\begin{align*}
\mathbf{r} &= \hat{B}^+ [\mathbf{M}(\dot{\mathbf{x}}^{\text{des}} + K_v \dot{e} + K_p e) + \mathbf{h}] + (\mathbf{I}_2 - \hat{B}^+ \hat{B})y \quad (a) \\
\mathbf{J}^T \mathbf{F}_1 &= - (\mathbf{M} \ddot{\mathbf{q}} + \mathbf{h}) + \mathbf{B} \mathbf{r} \quad (b) \\
\mathbf{F}_1 + \mathbf{F}_2 &= \mathbf{M}_0 \ddot{x}_b + \mathbf{g}_0 \quad (c) \\
\mathbf{A}_q \ddot{\mathbf{q}} + \mathbf{A}_x \dot{x}_b &= \mathbf{b} \quad (d) \\
\tilde{\mathbf{M}} \ddot{\mathbf{X}} + \tilde{\mathbf{h}} &= \tilde{\mathbf{B}} \mathbf{r} \quad (e) \\
[1 \quad \sigma \mu_1 \text{sign}(\dot{x}_b^{\text{des}})] \mathbf{F}_1 &= 0, \quad (f)
\end{align*}
\]

where \( \mathbf{F} = [\mathbf{F}_1 \mathbf{F}_2] \).

\( \sigma \leq 1 \) ensures \( \mathbf{F}_1 \) remains inside its related friction cone and ensure no slipping in the upper surface. The analytic solution for this optimization problem is given in Appendix D.

In phase 3, the second row of \( \tilde{\mathbf{B}} \) becomes zero and the equations of motion are as follows:

\[
\begin{align*}
\ddot{m}_1 \ddot{x}_s + \ddot{h}_1 &= \tilde{b}_{11} \tau_1 + \tilde{b}_{12} \tau_2 \\
\ddot{x}_b &= 0.
\end{align*}
\] (47)

Now, if \( \mathbf{r} \) is calculated similar to phase 1 and instead of \( \mathbf{D}_2 = 0 \), which causes singularity in \( \tilde{\mathbf{B}} \), \( \mathbf{D}_1 = 0 \) and \( \mathbf{D}_2 = 0 \), the control law similar to phase 1 increases the normal force on the object. Therefore after a limited period of time it can put the object on motion. This phenomenon can be described as follows. Since the object is in a static condition:

\[
\begin{align*}
|\Delta \mathbf{F}_{1y}| &= |\Delta \mathbf{F}_{2y}| \\
|\Delta \mathbf{F}_{1x}| &= \mu_1 |\Delta \mathbf{F}_{1y}| \\
|\Delta \mathbf{F}_{1x}| &= |\Delta \mathbf{F}_{2x}|.
\end{align*}
\] (48)

The physical condition for object coefficients of friction, \( \mu_1 > \mu_2 \), ensures that the contact force between object and horizontal surface moves to the edge of the friction cone and brings the object to the motion starting position. In this position, \( \mathbf{D}_2 \) is not zero any more and practically the controller switches to phase 1.

In phase 4, \( \tilde{\mathbf{B}} \) becomes zero, i.e., \( \mathbf{D}_1 = 0 \) and \( \mathbf{D}_2 = 0 \). Similar to phase 3, the object does not move and the controller must try to regulate the actuators such that the object leaves the static condition. In a similar way, if \( \mathbf{D}_1' = [1 \quad \mu_1 \text{sign}(\dot{x}_b^{\text{des}})] \) and \( \mathbf{D}_2' = [1 \quad \mu_2 \text{sign}(\dot{x}_b^{\text{des}})] \) are used instead of \( \mathbf{D}_1 = 0 \) and \( \mathbf{D}_2 = 0 \), the object is pushed to the motion starting position and the system switches to the other phases 1 or 2.

There are two points that must be mentioned here regarding this proposed approach. (i) There might be some cases that \( \mathbf{D}_1 = \mathbf{D}_2 = [1 \quad 0] \). In this case the sign of \( \dot{x}_s \) and \( \dot{x}_b \) changes simultaneously and \( \tilde{\mathbf{B}} \) is once again singular. This is not the...
case that we should worry about, since it rarely occurs and the system does not stay in this condition. (ii) It is clear that we need feedback of acceleration to compute $\alpha_i$ and $\beta_i$ \((i = 1, 2, 3)\) in the control law.

5. Numerical Results

Numerical results for both the conventional and proposed control approaches are given and compared in this section. The physical parameters listed in Table 2 are used for numerical purposes.

The object is assumed to track the following desired trajectory:

$$
\ddot{x}_b = \begin{cases} 
0.0256 & 0 < t < 1 \\
0 & 1 \leq t < 6 \\
-0.0256 & 6 \leq t < 7 
\end{cases} \tag{49}
$$

$$
x_b(0) = 1.366, \quad \dot{x}_b(0) = 0.
$$

In order to simulate the slipping phenomenon, we assume that during motion, the coefficient of friction of object and horizontal surface changes from its nominal value:

$$
\mu_2 = \bar{\mu}_2 \quad \text{if } 0 \leq t \leq 0.5 \text{ and } t > 2
$$

$$
\mu_2 = 0.15 \quad \text{if } 0.5 < t \leq 2. \tag{50}
$$

Note that control laws are calculated using nominal values.

Accuracy of the results of dynamical modeling is shown in Fig. 7 by plotting the difference between the results of the present model and the SimMech model for $x_b$, $\dot{q}_1$ and $\dot{q}_2$.

Superiority of the second tracking strategy (21) to the first one (20) for slippage control is shown in Fig. 8 by comparing the results of two strategies for $\dot{x}_s$ and $\dot{x}_b$.

Performances of the two approaches are compared in Fig. 9. The controller parameters in both approaches are similar.

Robustness of the controllers is studied numerically with reducing the mass and length parameters by 20% in the controller with respect to those of the system. Results are shown and compared in Fig. 10. It can be seen that the controller performance in the conventional approach is affected seriously, while the proposed approach can adjust itself to the new conditions very well.

<table>
<thead>
<tr>
<th>$m_1$</th>
<th>$m_2$</th>
<th>$L_1$</th>
<th>$L_2$</th>
<th>$m_b$</th>
<th>$\bar{\mu}_1$</th>
<th>$\bar{\mu}_2$</th>
</tr>
</thead>
<tbody>
<tr>
<td>1 kg</td>
<td>1 kg</td>
<td>1 m</td>
<td>1 m</td>
<td>2.5 kg</td>
<td>0.25</td>
<td>0.1</td>
</tr>
</tbody>
</table>
6. Experimental Setup and Results

An experimental setup is established to verify the method in practice. The setup and its model, in INVENTOR, are shown in Fig. 11. It consists of a two-link manipulator, a slider block, a slider railway, a control board and a longitudinal motion measurement encoder (1–5 in Fig. 11a). A flat surface that can freely rotate with respect to the second link is used as the end-effector in order to satisfy the non-rolling condition (Fig. 11b). In order to enforce the slippage that happens in the experiment, the railway is prepared such that its coefficient of friction with the slider block is changed in midway ($35 < x_b < 40$). The end-effector position and its derivatives are measured using joint sensors. Similar measurement for the slider is carried out utilizing the longitudinal motion encoder.
Figure 9. Comparison of results obtained from two control approaches. (a) Tracking error. (b) Object velocity. (c) End-effector movement on the object surface. (d) Sliding velocity of the end-effector on the object surface. (e) Time history of $\tau_1$. (f) Time history of $\tau_2$. 
Figure 10. Comparison of results obtained from two different control approaches when the model parameters differ from the actual parameters. (a) Tracking error. (b) Object velocity. (c) End-effector movement on the object surface. (d) Sliding velocity of the end-effector on the object surface. (e) Time history of $\tau_1$. (f) Time history of $\tau_2$. 
The following trajectory is set as the desired trajectory for the slider:

\[
\begin{align*}
\dot{x}_b^{\text{des}} &= \begin{cases} 
-0.0256 & 0 < t < 1 \\
0 & 1 \leq t < 6, \\
0.0256 & 6 \leq t < 7 
\end{cases}, \\
x_b^{\text{des}}(0) &= 0, \\
x_b^{\text{des}}(0) &= -0.27.
\end{align*}
\]  

The physical parameters of the system are given in Table 3.

Figure 12 compares the experimental results with those of the model for the above-mentioned parameters. As it is seen, the end-effector slides on the block in the initial and middle phases of motion. The initial phase slippage is due to the difference between parameters in the controller and the system, while the middle slippage is due to the changes in the friction coefficient between the slider and its railway. The controller stops these slippages and tries to perform the tracking objective in the remaining time. The slider velocity obtained from the simulation and experiment are compared with the desired one in Fig. 12a. Figure 12b compares the end-effector slippage velocity on the block. The motor torques calculated in the simulation are compared with those obtained from the experiment in Fig. 12c and 12d. Note that the experimental results are filtered by a low-pass filter in MATLAB. Figure 12e and 12f shows the effects of this filtering on the experimental results for block velocity as well as end-effector slippage velocity. The performance of the controller in controlling the end-effector slippage in practice and the accuracy of the dynamical modeling is clearly seen from these experimental results.
Figure 12. Comparison of experimental and simulation results. (a) Object velocity. (b) End-effector slippage velocity. (c) Time history of $\tau_1$. (d) Time history of $\tau_2$. (e) Filtered and unfiltered object velocity. (f) Filtered and unfiltered end-effector slippage velocity.

7. Conclusions

The sliding phenomenon in grasping an object by a manipulator is studied in this paper, considering a typical problem of moving an object by a manipulator. In or-
order to formulate and simulate the dynamics of the system, equality and inequality
equations of contact conditions are replaced by a single second-order differential
equation with switching coefficients. This kind of formulation allows us to synthe-
size the controller analytically. The accuracy of the model is verified by comparing
its results with those of SimMech of MATLAB software.

The conventional control method in grasping of an object is modified for the
cases that the end-effector of the manipulator slides on the object by including
the movement of the end-effector on the object and its velocity in the control law.
Although the modified method can simulate and control sliding of the manipulator
on the object, it does not show good performance, especially in the case that the
model parameters differ from the actual parameters. Since the method is a hybrid
feedforward and feedback control method, it is very difficult to prove the stability
of the method.

A new method is proposed by reducing the dynamical model to an input–output
reduced form. The governing equation in the new form is in the conventional
form, \( \tilde{M}\ddot{x} + \tilde{h} = \tilde{B}\tau \), with state-dependent matrices and vectors switching from
one phase of motion to another. It is shown that any stable control law for this
model guarantees internal stability of the system. In order to control the slippage
of the end-effector on the object and force the object to track the desired trajectory,
a multi-phase controller is proposed.

Performance and robustness of the proposed method are numerically compared
with the conventional method. Also, an experimental setup is established and
used to evaluate the controller performance practically. Both simulation and ex-
perimental results show the excellent performance of the controller in controlling
end-effector slippage while it still pushes the object to track its desired trajectory.
Although the convergence speed was not discussed, it can be changed by changing
the controller parameters, such as \( K_p \) and \( K_v \). Practically, this convergence speed
can be realized from simulation results, as well as from experimental results.

References

1. J. K. Salisbury and B. Roth, Kinematic and force analysis of articulated mechanical hands, ASME
2. B. Mishra, J. T. Schwartz and M. Sharir, On the existence and synthesis of multifinger positive
4. Y. H. Liu, Qualitative test and force optimization of 3-D frictional form closure grasps using linear
5. Y. Zheng and W. H. Qian, An enhanced ray-shooting approach to force-closure problems,
6. L. Han, J. C. Trinkle and Z. X. Li, Grasp analysis as linear matrix inequality problems, IEEE
7. Y. C. Park and G. P. Starr, Grasping synthesis of polygonal objects using a three-fingered robot


Appendix

A. Values of $\alpha_i$ and $\beta_i$ ($i = 1, 2, 3$)

Table A.1. Values for $\alpha = [\alpha_1 \ \alpha_2 \ \alpha_3]$ and $\beta = [\beta_1 \ \beta_2 \ \beta_3]$ under different conditions

<table>
<thead>
<tr>
<th>$\dot{x}_b$</th>
<th>$\dot{x}_s = 0$</th>
<th>$\dot{x}_s \neq 0$</th>
<th>$\dot{x}_b = 0$</th>
<th>$\ddot{x}_b = 0$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Movement</td>
<td>$\alpha^T = [0 \ 1 \ \text{sign}(\dot{x}_s)]$</td>
<td>$\alpha^T = [0 \ 1 \ 0]$</td>
<td>$\alpha^T = [1 \ 0 \ 0]$</td>
<td>$\alpha^T = [0 \ 1 \ 1]$</td>
</tr>
<tr>
<td>Motion</td>
<td>$\beta^T = [0 \ 1 \ 0]$</td>
<td>$\beta^T = [0 \ 1 \ 0]$</td>
<td>$\beta^T = [0 \ 1 \ 0]$</td>
<td>$\beta^T = [0 \ 1 \ 0]$</td>
</tr>
<tr>
<td>No Motion</td>
<td>$\alpha^T = [0 \ 1 \ 0]$</td>
<td>$\alpha^T = [0 \ 1 \ 0]$</td>
<td>$\alpha^T = [0 \ 1 \ 0]$</td>
<td>$\alpha^T = [0 \ 1 \ 0]$</td>
</tr>
<tr>
<td>Start forward</td>
<td>$\beta^T = [1 \ 0 \ 0]$</td>
<td>$\beta^T = [1 \ 0 \ 0]$</td>
<td>$\beta^T = [1 \ 0 \ 0]$</td>
<td>$\beta^T = [1 \ 0 \ 0]$</td>
</tr>
<tr>
<td>Start backward</td>
<td>$\beta^T = [0 \ 1 \ 0]$</td>
<td>$\beta^T = [0 \ 1 \ 0]$</td>
<td>$\beta^T = [0 \ 1 \ 0]$</td>
<td>$\beta^T = [0 \ 1 \ 0]$</td>
</tr>
</tbody>
</table>
B. Error Analysis for Strategies in (20) and (21)

Let us define $e_g = q_c - q$ and use a typical feedback linearization form for position control:

$$B \tau_e = M(q_c + K_{vq} \dot{e}_q + K_{pq} e_q) + h.$$  \hfill (B.1)

Substituting (B.1) and (19) in (1) we can write:

$$M \ddot{q} + h = M(q_c + K_{vq} \dot{e}_q + K_{pq} e_q) + h + J^T(F^{des}_{1} - F_{1}).$$  \hfill (B.2)

Hence, the governing equation of the tracking error is obtained as:

$$\ddot{e}_q + K_{vq} \dot{e}_q + K_{pq} e_q = -M^{-1}J^T e_{F_1},$$  \hfill (B.3)

assuming no uncertainty in the modeling.

In the first strategy, (20), since $x^{des}_c = x_b + x_s$, at each instant $q_c = q$. Hence equation (B.3) is rewritten as:

$$\ddot{e}_q + K_{vq} \dot{e}_q + K_{pq} e_q = -M^{-1}J^T e_{F_1}.$$  \hfill (B.4)

For the second strategy $q_c$ is not necessarily equal to $q$ and the tracking error is computed from (B.3). Comparing (B.3) and (B.4) shows the advantage of the second strategy. For example, once we have a constant term in the right-hand side of (B.3) and (B.4) in the steady-state phase of the system, (B.4) results in constant error in $\dot{x}_s$, while $\dot{x}_s$ vanishes according to (B.3) and (21).

C. Proof of Theorem 2

Notice that columns or rows of $\tilde{M}$ are functions of time and according to theorem of [28], $\tilde{M}$ is full row rank and then invertible on $[t_1 \ t_2]$, if and only if:

$$\Psi = \int_{t_1}^{t_2} \tilde{M} \tilde{M}^T \, dt,$$  \hfill (C.1)

is a non-singular matrix.

$\Psi$ is the Hermitian matrix, so it is a symmetric semi-positive definite matrix. Thus, if is also non-singular, it will be positive definite and regarded to Rayliegh–Ritz theorem:

$$\lambda_1 I_2 \leq \Psi \leq \lambda_2 I_2,$$  \hfill (C.2)

where positive scalars $\lambda_1$ and $\lambda_2$ are minimum and maximum eigenvalues of $\Psi$, respectively, and vary with time $t_1$ and $t_2$. If we choose $t_1 = t_0$ and $t_2 = t_0 + T$ and define $\beta_1 = \sup_{t_0 \in (-\infty, +\infty)} \lambda_1(t_0, t_0 + T)$ and $\beta_2 = \inf_{t_0 \in (-\infty, +\infty)} \lambda_2(t_0, t_0 + T)$, then $\beta_1$ and $\beta_2$ are constant scalars and we can write for all $t_0$:

$$\beta_1 I_2 \leq \int_{t_0}^{t_0+T} \tilde{M} \tilde{M}^T \, dt \leq \beta_2 I_2.$$  \hfill (C.3)
Since we have shown that velocity and position tracking errors converge to zero, the above condition will be met if the desired trajectory satisfies:

\[
\beta_1 I_2 \leq \int_{t_0}^{t_0+T} \tilde{M}_d \tilde{M}_d^T dt \leq \beta_2 I_2,
\]

where \( \tilde{M}_d = \tilde{M}(x_b^{\text{des}}, x_s^{\text{des}}) \).

**D. Analytical Solution to (46)**

Using (46e) we can write:

\[
\ddot{x}_b = K_b \tau + \lambda_b,
\]

where \( K_b \) and \( \lambda_b \) are the first row of \( \tilde{M}^{-1} \tilde{B} \) and \( -\tilde{M}^{-1} \tilde{h} \), respectively. Substituting (D.1) in (46d), \( \ddot{q} \) is calculated as:

\[
\ddot{q} = -A_q^{-1} \dot{A}_q A_b \tau - A_q^{-1} A_q \lambda_b + A_q^{-1} \dot{b}.
\]

Substituting (D.1) and (D.2) in (46b) and (46c), we obtain:

\[
F_1 = K_1 \tau + \lambda_1, \quad F_2 = K_2 \tau + \lambda_2,
\]

where:

\[
K_1 = J^{-T} (B + MA_q^{-1} A_q K_b), \quad K_2 = M_0 K_b - K_1 \quad \lambda_1 = J^{-T} M A_q^{-1} A_q \lambda_b - J^{-T} M A_q^{-1} (1 + J^{-T} \dot{h}), \quad \lambda_2 = M_0 \lambda_b + g - \lambda_1.
\]

Defining \( S = \tilde{B}^+[\tilde{M}(\ddot{x}_b^{\text{des}} + K_v \dot{e} + K_p e) + \tilde{h}] \) and using (46a), (D.3) can be rewritten as:

\[
F_1 = K_1' y + \lambda_1', \quad F_2 = K_2' y + \lambda_2',
\]

where:

\[
K_1' = K_1 (I_2 - \tilde{B}^+ \tilde{B}), \quad K_2' = K_2 (I_2 - \tilde{B}^+ \tilde{B}) \quad \lambda_1' = K_1 S + \lambda_1, \quad \lambda_2' = K_2 S + \lambda_2.
\]

If we define:

\[
K_f = \begin{bmatrix} K_1' \\ K_2' \end{bmatrix}, \quad \lambda_f = \begin{bmatrix} \lambda_1' \\ \lambda_2' \end{bmatrix},
\]

and use (46f), then (D.6) can be reformulated as:

\[
Q \begin{bmatrix} F_{1Y} \\ F_{2X} \\ F_{2Y} \end{bmatrix} = K_f y + \lambda_f, \quad \text{where } Q = \begin{bmatrix} -\sigma_1 \mu_1 \text{sign}(\ddot{x}_b^{\text{des}}) & 0 & 0 \\ 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}.
\]

As one can see, \( Q \) is full column rank, so:

\[
\begin{bmatrix} F_{1Y} \\ F_{2X} \\ F_{2Y} \end{bmatrix} = Q^+ K_f y + Q^+ \lambda_f.
\]
Now if we choose \( y \) as:
\[
y = - (Q^+ K_f)^+ Q^+ \lambda_f,
\]
(D.11)

\( F \) will have its minimum norm with respect to \( y \) \[29\].

**About the Authors**

**Shahram Hadian Jazi** received his BS and MS degrees in Mechanical Engineering in 1998 and 2000, respectively, both from Isfahan University of Technology (IUT), Isfahan, Iran, where he is currently working towards his PhD degree. His research interests are robotics (analysis, design and manufacturing) and control of dynamical systems. He is an active member of the Dynamic and Robotic Research Group of IUT and has been involved in several projects of the group. He is a member of the ASME.

**Mehdi Keshmiri** received his BS and MS degrees in Mechanical Engineering, both from Sharif University of Technology, Tehran, Iran, in 1986 and 1989, respectively, and his PhD from McGill University, Montreal, Canada, in 1995. Since 1995, he has been with the Department of Mechanical Engineering at Isfahan University of Technology, Isfahan, Iran. His research interests are system dynamics, control of dynamical systems and vibration, and also he is very active in the field of science park development and technology management.

**Farid Sheikholeslam** received his BS degree in Electronics from Sharif University of Technology, Tehran, Iran, in 1990. He also received his MS degree in Communications and a PhD in Electrical Engineering from Isfahan University of Technology (IUT), Isfahan, Iran, in 1994 and 1998, respectively. Since 1999 he has been with the Department of Electrical and Computer Engineering at IUT. His research interests are robotics, control algorithms, fuzzy systems, neural networks, stability analysis and nonlinear systems.