Multibody Syst Dyn (2018) 44:163–189
https://doi.org/10.1007/s11044-018-9617-8

Ball-and-finger system: modeling and optimal trajectories

M. Mahdi Ghazaei Ardakani[1,2] · Fredrik Magnusson[2]

Received: 27 April 2017 / Accepted: 7 December 2017 / Published online: 25 January 2018

Abstract A rigid-body model of a finger interacting with a trackball is considered. The proposed system is a suitable candidate for studying trajectory generation when interaction plays an important role, such as in assembly and manipulation tasks. The mathematical model consists of a ball with a spherical joint constraint, a finger with three degrees of freedom, and the Coulomb friction model. From first principles, we derive a hybrid, high-index differential-algebraic equation for modeling the system dynamics, which is used for both simulation and finding optimal trajectories. For this problem, task planning, path planning, and trajectory generation are strongly interrelated, which makes using an integrated approach to trajectory generation inevitable. Moreover, the trajectory generation algorithm has to handle a number of important features, e.g., unilateral and non-holonomic constraints.

Keywords Hybrid dynamic optimization · Optimal trajectory · Dynamics with contacts · Non-holonomic constraint

1 Introduction

Model-based design is the prevailing approach in engineering. The automotive and aerospace industries make extensive use of modeling and optimization tools to improve their products. As the available computation power increases, so too does the viability of using model-based optimization to solve control problems.

Practical systems often contain a combination of discrete and continuous state variables. Accurate modeling of such systems requires the incorporation of hybrid dynamics. Hybrid models often appear in the context of processes with switched dynamics. For example in

✉ M. Mahdi Ghazaei Ardakani
mahdi.ghazaei@iit.it

F. Magnusson
bleevoe@gmail.com

[1] Department of Advanced Robotics (ADVR), Istituto Italiano di Tecnologia (IIT), Via Morego 30, 16163 Genoa, Italy

[2] Department of Automatic Control, LTH, Lund University, 221 00 Lund, Sweden

mechanics, establishing or breaking contact changes the behavior of the system. Although it is possible to model the rapid changes in forces resulting from such phenomena, the resulting differential equations are stiff and cause trouble for numerical methods. An alternative approach is to instead allow discontinuities in the state trajectories. The framework of hybrid systems allows both switched dynamics and discontinuous state variables.

Applying optimization techniques to hybrid models has resulted in the solution of various practical problems. In locomotion planning, there are many examples where hybrid dynamic optimization methods have been used. For example, gait generation for bipedal and quadrupedal robots is addressed in [17]. In [12], the full dynamics of a humanoid is considered to find a walking pattern. Hybrid dynamic optimization is also used in other fields of engineering, e.g., to make discrete decisions in power plants [13].

A competing paradigm to the model-based approach is the model-free approach. Using a variety of model-free techniques, successful results for optimal control of robots interacting with the environment have been reported [21, 30, 32]. Nevertheless, model-free approaches provide us with little or no insight about fundamental limitations and difficulties.

This paper contributes by presenting a simple, yet realistic enough, example system to capture the main issues arising in modeling and optimal control of dynamics with varying contacts. We propose a ball and finger system resembling a trackball interacted with by a human finger. *Unilateral* and *non-holonomic* constraints play an important role in this system. In order to derive analytic equations, simple 3D geometries are considered. The model possesses the following features:

- Analytic 3D models with contacts
- Rolling of finger against ball
- Dynamics and slippage
- Suitable for simulation as well as optimization

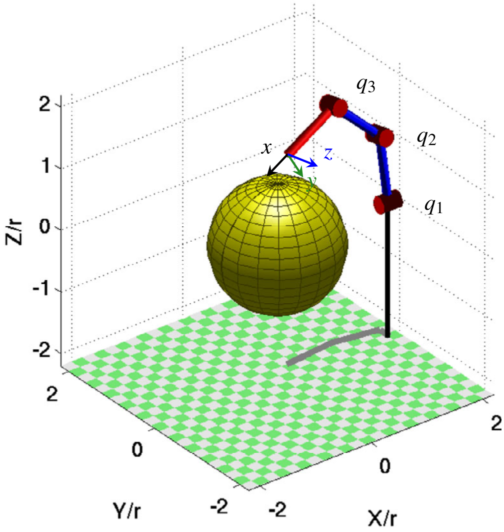
There are elaborate mathematical models for studying contact stability [35, 36]. In contrast, the purpose of the model developed here is to capture contact transitions and to find optimal trajectories. The hybrid dynamic optimization required for this purpose has been performed utilizing a multiphase approach, i.e., a sequence of the discrete modes of the system is mapped to phases.

The rest of the paper is organized as follows. In Sect. 2, the mechanical model of the process is developed. Based on this model, we construct a hybrid system in Sect. 3, which is later on used for simulation as well as optimization. Section 4 discusses the trajectory-planning problem and the optimization approach to find optimal solutions. Various tools have been used for simulation and optimization. The tool chain is detailed in Sect. 5. We introduce instances of the trajectory-planning problems and give the results of the optimization and simulation in Sect. 6. Section 7 discusses various aspects of the modeling and optimization. We draw conclusions and propose future research in Sect. 8. Additionally, Appendices A and B provide the details of the mathematical expressions for the kinematics and the dynamics of the finger, respectively.

2 Mechanical modeling

In this section, we build the mechanical model of the process shown in Fig. 1. The ball is positioned in a socket joint which is not shown in the figure. To produce analytic equations, we consider an idealized model, which only admits contact between a sphere and a line segment. The following list summarizes the modeling assumptions:

Fig. 1 An illustration of the ball-and-finger system for $\ell = r$



1. The finger is modeled as a rigid body with three degrees of freedom (DOF).
2. The ball is a sphere with radius r, which is held by a socket imposing a spherical joint constraint.
3. The Coulomb friction model is considered with static (μ_s) and kinetic (μ_k) friction constants.
4. There is viscous friction in the joints and between the socket and the ball, with friction constants μ_v and D, respectively.
5. There is no torsional friction.
6. The collision between the finger and the ball is inelastic, i.e., when the finger collides with the ball it will not bounce back. Accordingly, the coefficient of restitution is equal to zero.
7. The links are ideal line segments.
8. In case of contact, single point contact either at the end-effector or along the last link is allowed. This is guaranteed by the choice of the geometry.
9. The contact points are non-adhesive.
10. The centers of gravity (COG) are located at the middle of each link and inertia tensors with respect to these points are diagonal.

2.1 Geometry and kinematics

Considering Fig. 1, the geometry of the system is described here. The ball with radius r is assumed to be located at the origin $o_b = [0 \;\; 0 \;\; 0]^T$. The base of the finger is specified by the transformation consisting of a translation $o_f = [2r \;\; 0 \;\; 0]^T$ and a rotation -90 degrees around the y-axis. The kinematics of the finger is specified by the Denavit–Hartenberg (DH) parameters [10] according to Table 1. We use $q \in \mathbb{R}^3$ to denote the finger joint angles and the length of every link is $\ell \leq r$. Knowing the DH parameters and the base transformation, the forward kinematics and the geometric Jacobian with respect to the fingertip can be calculated. See Appendix A for details.

Table 1 DH parameters of the finger

Link	a_i	α_i	d_i	θ_i
1	ℓ	$\pi/2$	0	q_1
2	ℓ	0	0	q_2
3	ℓ	0	0	q_3

2.2 Collision

To detect collision between the ball and the finger, the closest distance between the ball and the last link is found. Define $p_m := p_e + \alpha \hat{e}_{x_3}$ as the point on the line corresponding to the last link closest to the ball, where $\hat{e}_{x_3}$ is the unit vector parallel to link 3, p_e is the position of the fingertip, and α determines how far p_m lies from the fingertip. Then

$$(p_m - o_b) \cdot \hat{e}_{x_3} = 0, \tag{1}$$

which results in

$$\alpha = (o_b - p_e) \cdot \hat{e}_{x_3}. \tag{2}$$

Define also the potential collision point, i.e., the point on link 3 closest to the ball

$$p_c := p_e + \mathrm{sat}_{-a_3}^{0}(\alpha)\hat{e}_{x_3}, \tag{3}$$

where $\mathrm{sat}_L^U(x)$ denotes the saturation function with L as the lower limit and U as the upper limit and $a_3 = \ell$ is the length of the third link. Hence, the signed shortest distance from the third link to the ball is

$$h(q) := \|r_c\| - r, \tag{4}$$

where $r_c := p_c - o_b$ is the vector to the closest point on the finger to the ball.

If the ball and the finger are impenetrable, it is required that $h(q) \geq 0$. Therefore, the condition for collision can be stated as

$$h(q) \leq 0. \tag{5}$$

The velocity of the third link at the potential contact point denoted by v_c can be derived from the velocity of the end point v_e using the velocity relation of rigid motion [26] according to

$$\begin{aligned} v_c &:= v_e + \omega_3 \times (p_c - p_e) \\ &= v_e - \mathrm{sat}_{-a_3}^{0}(\alpha)\hat{e}_{x_3} \times \omega_3 \\ &= \big(J_p - \mathrm{sat}_{-a_3}^{0}(\alpha) S(\hat{e}_{x_3}) J_o\big)\dot{q} \\ &= J_c \dot{q}, \end{aligned} \tag{6}$$

where ω_3 is the angular velocity of the last link and $S(\cdot)$ denotes the skew-symmetric operator performing the cross-product. We have also defined

$$J_c := J_p - \mathrm{sat}_{-a_3}^{0}(\alpha) S(\hat{e}_{x_3}) J_o, \tag{7}$$

which is the Jacobian at the potential contact point. Note that v_c is the velocity of the link at the contact point and not the velocity of the contact point itself, i.e., $v_c \neq \dot{r}_c$.

2.3 Constrained motion

When the finger is touching the ball, its motion is constrained to a manifold. Using (4), the finger is in contact with the ball when

$$h(q) = 0. \tag{8}$$

This is equivalent to

$$\|r_c\|^2 - r^2 = 0. \tag{9}$$

Differentiating (9) w.r.t. time results in

$$v_c \cdot r_c = 0. \tag{10}$$

Hence, when in contact, the velocity of the third link at the contact point remains tangential to the ball.

Depending on whether there is an interaction force between the objects, the contact may be active or inactive. Let us denote the normal vector at the contact point outwards w.r.t. the ball by

$$n := \frac{r_c}{\|r_c\|}. \tag{11}$$

The normal component of the interaction force at the contact is then given by

$$\lambda_n := \lambda \cdot n, \tag{12}$$

where λ denotes the contact force exerted on the finger. Using the Coulomb friction model [8], interaction forces between two bodies are possible if and only if there is a normal force between the contacting surfaces. Therefore, an active contact implies $\lambda_n > 0$.

The relative velocity between the finger and the ball at the contact point is

$$\Delta v := \omega \times r_c - v_c, \tag{13}$$

where ω is the angular velocity of the ball. When two bodies are sticking, the relative velocity between them is zero, i.e.,

$$\Delta v = 0. \tag{14}$$

This constraint is maintained by the static friction.

At this stage, it is useful to introduce $f_\perp$ and $f_\parallel$ for the perpendicular and tangential decomposition of an arbitrary vector f w.r.t. a plane with normal vector n, according to

$$f = f_\parallel + f_\perp, \tag{15a}$$

$$f_\perp = (v \cdot n)n = (n \otimes n)f = P_n f, \tag{15b}$$

$$f_\parallel = -n \times (n \times f) = f - f_\perp = (1 - P_n)f = T_n f, \tag{15c}$$

where $\otimes$ denotes the outer product, 1 is the identity matrix of proper dimension, and we have defined

$$P_n := (n \otimes n), \tag{16}$$

$$T_n := 1 - P_n. \tag{17}$$

Now, let us define

$$\sigma := \mu_s \lambda_n - \|\lambda_\parallel\|, \tag{18}$$

where $\lambda_\parallel$ is the force resulting from friction. Hence, according to the Coulomb model, the condition for stiction is

$$\|\lambda_\parallel\| < \mu_s \lambda_n \Leftrightarrow \sigma > 0. \tag{19}$$

When the finger is slipping against the ball, the relative velocity Δv is not zero anymore and

$$\lambda_\parallel = \mu_k \lambda_n \frac{\Delta v}{\|\Delta v\|}. \tag{20}$$

Note that in this case Δv is tangential to the ball surface, since using the definition (13) and (10) it is possible to show

$$\Delta v \cdot r_c = (\omega \times r_c) \cdot r_c - v_c \cdot r_c = 0. \tag{21}$$

The contact is inactive when $\lambda_n \leq 0$.

Depending on the contact point between the finger and ball, two types of contact may arise: point and sphere or line and sphere. If the contact is not slipping, for the former type of contact, (14) can be integrated to result in a geometrical constraint, while the latter type yields a rolling without slipping constraint which is non-holonomic.

2.4 Dynamics

In this section, we derive the equations of motion for the ball and finger system. The kinetic and potential energy of the finger can be described by

$$\mathcal{T}_f = \sum_{i=1}^{3} \frac{1}{2} m_i \dot{q}^T \left(J_P^i\right)^T J_P^i \dot{q} + \sum_{i=1}^{3} \frac{1}{2} \dot{q}^T \left(J_O^i\right)^T R_i I_i R_i^T J_O^i \dot{q}, \tag{22a}$$

$$\mathcal{U}_f = \sum_{i=1}^{3} m_i g_0^T p_i, \tag{22b}$$

respectively. Here, m_i and I_i denote the mass and inertia matrix of link i, respectively, J_P^i and J_O^i denote partial Jacobians up to link i, R_i is the rotation matrix of link i with respect to the world coordinate, $g_0 = [0 \;\; 0 \;\; -g]^T$ is the gravity acceleration vector, and p_i denotes the position vector to the center of the gravity of link i. The Lagrangian of the finger is $\mathcal{L}_f := \mathcal{T}_f - \mathcal{U}_f$. Accordingly, we derive the equations of motion [7]

$$\frac{\mathrm{d}}{\mathrm{d}t}\left(\frac{\partial \mathcal{L}_f}{\partial \dot{q}}\right) + \frac{\partial \mathcal{L}_f}{\partial q} = Q^e + Q_f^c. \tag{23}$$

The generalized external forces and the generalized forces due to the kinematic constraints are denoted by Q^e and Q_f^c, respectively. Substituting Q^e with the actuation torques, τ, minus the viscous friction, we obtain

$$M_f(q)\ddot{q} + C(q, \dot{q})\dot{q} + \mu_v \dot{q} + g(q) = \tau + Q_f^c, \tag{24}$$

where $M_f(q)$ is the mass matrix of the finger, $C(q,\dot{q})\dot{q}$ denotes the contribution of the centrifugal and Coriolis forces, μ_v is the viscous friction coefficient, and $g(q)$ is the gravitational force. The expressions for these functions are given in Appendix B.

For the ball, using the Newton–Euler equations, we find

$$I_b\dot{\omega} + \omega \times I_b\omega + D\omega = Q_f^c, \tag{25}$$

where $I_b = \frac{2}{5}mr^2\mathbb{1}$ is the inertia tensor for the ball and D denotes the friction constant with the socket.

In the rest of this section, we consider different regions depending on the active constraints imposed by the contact. The effect of collision, i.e., switching between no contact and contact, is also considered. Finally, we describe the overall system.

2.4.1 Free motion

In case of no contact, we have $Q_f^c = 0$. Hence,

$$M_f(q)\ddot{q} + C(q,\dot{q})\dot{q} + \mu_v\dot{q} + g(q) = \tau, \tag{26a}$$

$$I_b\dot{\omega} + \omega \times I_b\omega + D\omega = 0. \tag{26b}$$

2.4.2 Sticking

The condition of no slipping (14) can be rewritten as

$$G\begin{bmatrix}\dot{q}\\ \omega\end{bmatrix} = 0, \tag{27}$$

where we have defined $G := [J_c\ S(r_c)] \in \mathbb{R}^{3\times 6}$. The generalized forces due to the constraints according to the Lagrange–d'Alembert theorem [7] can be derived as

$$\begin{bmatrix}Q_f^c\\ Q_b^c\end{bmatrix} = G^T\lambda, \tag{28}$$

where λ are the Lagrange multipliers corresponding to the interaction forces. Therefore, (24) and (25) can be rewritten as

$$M_f(q)\ddot{q} + C(q,\dot{q})\dot{q} + \mu_v\dot{q} + g(q) = \tau + J_c^T\lambda, \tag{29a}$$

$$I_b\dot{\omega} + \omega \times I_b\omega + D\omega = -r_c \times \lambda. \tag{29b}$$

The interaction forces λ may be solved for in order to reduce the DAE into an ODE. They could be used to evaluate whether in a certain state sticking is possible without switching to new dynamics. The calculation may also be used to find out the direction of kinetic friction at the onset of slipping. Let us define the block diagonal matrix

$$M := \operatorname{blkdiag}\big(M_f(q), I_b\big). \tag{30}$$

We can rewrite (29a)–(29b) as

$$M\begin{bmatrix}\ddot{q}\\ \dot{\omega}\end{bmatrix} + f(q,\dot{q},\omega,\tau) = G^T\lambda, \tag{31}$$

where we have defined

$$f(q,\dot{q},\omega,\tau) := \begin{pmatrix} C(q,\dot{q})\dot{q} + \mu_v\dot{q} + g(q) - \tau \\ \omega \times I_b\omega + D\omega \end{pmatrix}. \tag{32}$$

The Lagrange multipliers λ can be calculated by first taking the derivative of the constraints (27). Afterward, we multiply (31) by GM^{-1} from the left and use the relation obtained from the derivative of the constraints. Let us define $\Gamma := GM^{-1}G^T$. Assuming Γ^{-1} exists, $\lambda = \lambda(t,q,\dot{q},\omega)$ can be calculated according to

$$\begin{aligned} \lambda &= \Gamma^{-1}\left(GM^{-1}f(q,\dot{q},\omega,\tau) - \sum_{k=1}^{n}\left(\frac{\partial G}{\partial q_k}\dot{q}_k\right)\begin{bmatrix}\dot{q}\\ \omega\end{bmatrix}\right) \\ &= \Gamma^{-1}\big(GM^{-1}f(q,\dot{q},\omega,\tau) + \omega \times \dot{r}_c - \dot{J}_c\dot{q}\big). \end{aligned} \tag{33}$$

Note that the matrix M is always invertible with

$$M^{-1} = \mathrm{blkdiag}\big(M_f^{-1}(q), I_b^{-1}\big). \tag{34}$$

2.4.3 Slipping

In the case of slipping, the constraint only defines the normal part of the interaction force, i.e., λ_n. By rewriting (10), we find

$$r_c \cdot J_c\dot{q} = 0. \tag{35}$$

We normalize (35) by dividing by $\|r_c\|$, to get

$$n^T J_c\dot{q} = 0. \tag{36}$$

Now, from the Lagrange–d'Alembert theorem, it is clear that the contribution of the constraint force is $Q_f^c = J_c^T n\lambda_n$. Including the kinetic friction forces specified by (20), we conclude

$$M_f(q)\ddot{q} + C(q,\dot{q})\dot{q} + \mu_v\dot{q} + g(q) = \tau + J_c^T(\lambda_\parallel + n\lambda_n), \tag{37a}$$

$$I_b\dot{\omega} + \omega \times I_b\omega + D\omega = -r_c \times \lambda_\parallel. \tag{37b}$$

It is also possible to reformulate (37a)–(37b) using the total interaction force λ such that we obtain the same relations as (29a)–(29b). In this case, (20) can be written as

$$\lambda \cdot \Delta v = \|\Delta v\|\mu_k(\lambda \cdot n), \tag{38a}$$

$$(n \times \Delta v) \cdot \lambda = 0, \tag{38b}$$

which additionally has no risk of division by zero as opposed to (20).

2.5 Impacts

Collisions give rise to rapid changes of the states. By allowing discontinuities in the state trajectory, it is possible to model the effect of impacts right after the collision without dealing with the actual process. In this case, new values for post-impact quantities must be calculated. Since these values can have jumps, we use superscripts + and − to denote the right and left limits of a quantity, respectively. Next, we consider various impact laws. Since after collision the finger is neither allowed to penetrate the ball nor to bounce off directly (coefficient of restitution equal to zero), all the impact models ensure $(J_c\dot{q}^+)_\perp = n \cdot J_c\dot{q}^+ = 0$.

2.5.1 Newton's impact law

According to Newton's impact law [28], after impact the tangential velocity of the finger remains intact and only its normal velocity is vanished because of the assumed inelasticity of the collision. Hence,

$$J_c\dot{q}^+ = (J_c\dot{q})_\parallel^- = T_n J_c\dot{q}^-. \tag{39}$$

Accordingly, we can solve for the post-impact joint velocities. In this case, the impulse is perpendicular to the ball and cannot change the velocity of the ball.

2.5.2 Sticking

When sticking occurs, the force impulse ι resulting from the impact makes the post-collision relative velocity $\Delta v^+ = 0$. From the equations of motion (29a)–(29b) and (13), we conclude

$$M_f(q)\big(\dot{q}^+ - \dot{q}^-\big) = J_c^T \iota, \tag{40a}$$

$$I_b\big(\omega^+ - \omega^-\big) = -r_c \times \iota, \tag{40b}$$

$$\omega^+ \times r_c - J_c\dot{q}^+ = 0, \tag{40c}$$

which can be solved for $\dot{q}^+$, ω^+, and ι. By combining (40a)–(40c), we find

$$\begin{aligned}\big(J_c M_f^{-1} J_c^T - S(r_c) I_b^{-1} S(r_c)\big)\iota &= \omega^- \times r_c - J_c\dot{q}^- \\ &= \Delta v^-.\end{aligned} \tag{41}$$

This shows that the impulse ι is a function of the contact point r_c, the configuration of the finger q, and the pre-impact relative velocity Δv^-, but neither the absolute velocity of the finger nor the ball.

2.5.3 Slipping

In the case of slipping, an impact law can be formulated as minimizing the post-impact energy, based on the maximum dissipation principle [16]. The optimization problem to be solved is

$$\begin{aligned}
&\text{minimize} && \frac{1}{2}\omega^{+T} I_b \omega^+ + \frac{1}{2}\dot{q}^{+T} M_f(q)\dot{q}^+, && (42a)\\
&\text{with respect to} && \iota, \omega^+, \dot{q}^+, && \\
&\text{subject to} && M_f(q)\big(\dot{q}^+ - \dot{q}^-\big) = J_c^T \iota, && (42b)\\
& && I_b\big(\omega^+ - \omega^-\big) = -r_c \times \iota, && (42c)\\
& && n \cdot J_c\dot{q}^+ = 0, && (42d)\\
& && \|\iota_\parallel\| \le \mu_k \iota \cdot n. && (42e)
\end{aligned}$$

A heuristic way is to minimize the post-impact relative velocity. This strategy allows a quick transition to sticking. The corresponding optimization problem is

$$\begin{aligned}
&\text{minimize} && \|\Delta v^+\|, && \text{(43a)}\\
&\text{with respect to} && \mu, \iota, \omega^+, \dot{q}^+, &&\\
&\text{subject to} && M_f(q)\big(\dot{q}^+ - \dot{q}^-\big) = J_c^T \iota, && \text{(43b)}\\
& && I_b\big(\omega^+ - \omega^-\big) = -r_c \times \iota, && \text{(43c)}\\
& && n \cdot J_c \dot{q}^+ = 0, && \text{(43d)}\\
& && \iota_\parallel = \mu \iota_n \frac{\Delta v_\parallel^-}{\|\Delta v_\parallel^-\|}, && \text{(43e)}\\
& && 0 < \mu \leq \mu_k, && \text{(43f)}
\end{aligned}$$

where we have additionally assumed that the direction of the tangential part of the impulse is aligned with the pre-impact relative velocity. Problem (43a)–(43f) can be solved analytically [15]. Subsequently, the equations can be used in an optimal control problem.

2.6 Overall system

The complete system has three discrete modes, corresponding to free motion, sticking, and slipping. Let λ denote the interaction force and $x^T := (q^T, dq^T, \omega^T)$, where $\dot{q} = dq$. If for the moment we do not consider the transition between modes, we can unify all the modes with the first-order differential equation

$$\begin{pmatrix} 1 & 0 \\ 0 & M(q) \end{pmatrix} \dot{x} + \begin{pmatrix} 0 & -1 & 0 \\ 0 & C(q, dq) + \mu_v & 0 \\ 0 & 0 & S(\omega) I_b + D \end{pmatrix} x + \begin{pmatrix} 0 \\ g(q) \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ \tau \\ 0 \end{pmatrix} + \begin{pmatrix} 0 \\ J_c^T(q) \\ -S(r_c(q)) \end{pmatrix} \lambda. \tag{44}$$

The determination of λ is separate for each mode as follows:

- Free motion,

$$\lambda = 0. \tag{45}$$

- Sticking (valid when $\sigma > 0$),

$$J_c \dot{q} + r_c \times \omega = 0. \tag{46}$$

- Slipping (valid when $\Delta v \neq 0$),

$$\begin{aligned}
r_c \cdot J_c \dot{q} &= 0, && \text{(47a)}\\
\lambda \cdot \Delta v &= \|\Delta v\| \mu_k \lambda_n, && \text{(47b)}\\
(n \times \Delta v) \cdot \lambda &= 0. && \text{(47c)}
\end{aligned}$$

Hence, we have a differential-algebraic equation (DAE) describing the dynamics in each mode, which is low index in the case of free motion and of index 2 in the case of sticking and slipping.

3 System modeling

The aim of this section is to build a hybrid dynamical system based on the mechanical system introduced in Sect. 2.6. To find the solution to the overall system, we must also consider an impact law and the so called Signorini's condition or corner law for λ_n (see [8]):

$$\begin{aligned} &h \cdot \lambda_n = 0, \\ &h \geq 0, \quad \lambda_n \geq 0. \end{aligned} \tag{48}$$

Firstly, we discuss the relation between these complementarity conditions and mode transitions. In the latter subsection, the complete hybrid system is presented.

3.1 Mode transitions

In view of the complementarity conditions (48), we can partition a state trajectory into three different regions:

$$\lambda_n = 0 \wedge h > 0, \tag{49a}$$

$$\lambda_n > 0 \wedge h = 0, \tag{49b}$$

$$\lambda_n = 0 \wedge h = 0. \tag{49c}$$

The first partition is a region where the dynamics are unconstrained. The second partition is related to active contacts where constrained dynamics are applicable. The third partition defines states, where there is a contact but the constraint is inactive. This set can be reached from either the unconstrained dynamics or the constrained dynamics.

Using the complementarity conditions, it is possible to determine whether certain dynamics can be assumed in a given region. However, for simulation of the dynamics forward, (48) does not provide explicit transition conditions between unconstrained and constrained modes. In continuous time a complete causal description of the transition condition is not generally possible. Thus, by allowing a small violation of the validity regions, the transition time can be approximated arbitrarily close to the actual ones.

Starting from the free motion where $\lambda_n = 0$, using the unconstrained dynamics when $h = 0$ and either $\dot{h} < 0$ or $\dot{h} = 0 \wedge \ddot{h} < 0$ leads to penetration. Therefore, to guarantee impenetrability new initial velocities must be calculated (in the case of inelastic collision, $\dot{h} = 0$). Accordingly, the transition from the free motion is triggered when

$$h < 0 \vee (h = 0 \wedge \dot{h} < 0) \vee (h = \dot{h} = 0 \wedge \ddot{h} < 0). \tag{50}$$

Depending on the post-impact conditions, the state trajectory can either switch to the constrained dynamics or continue evolving according to the unconstrained dynamics. On the other hand, starting from either of the constrained dynamics where $h = 0$, the transition to the free motion can be triggered when $\lambda_n < 0$.

The modes and the transitions are illustrated in Fig. 2, following the notation in [22]. Each mode is represented by a circle and can have its own set of dynamical equations. The

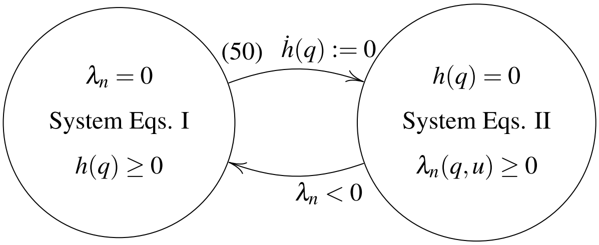


Fig. 2 Hybrid system representation of the complementarity condition (48)

conditions for the validity of a mode is given at the lowest part of the circle. The transition conditions are shown close to the tail of the arrows and the reset maps close to the arrowheads. Note that $h < 0$ and $\lambda_n < 0$ in the transition conditions should be interpreted as when the boundaries at zero are crossed. In the case of dynamic simulation, this means that the solver has to pinpoint when the transition condition becomes satisfied with equality and then perform the transition.

To account for switching to slipping and sticking and between them, we should also consider their region of validity. Both sticking and slipping constitute constrained modes where $h = 0$. The transition to them is, however, distinguished by the relative post-impact velocity Δv^+. If the static friction is not able to maintain the sticking constraint, i.e., according to (19) when $\sigma \leq 0$, the system makes a transition to slipping. In the opposite direction, the transition depends on the relative velocity. However, (20) is undefined for zero relative velocity. This implies that there is no ideal transition condition between the slipping and sticking modes, although it is possible to calculate a continuous solution for the velocities. This can be resolved by conditioning the transition on $\|\Delta v\| \leq \epsilon$. The switching then happens arbitrarily close to the actual transition by tuning the value of ϵ. To avoid switching directly back we should also make sure that $\mathrm{d}\|\Delta v\|/\,\mathrm{d}t < 0$.

Note that switching on $\|\Delta v\|$ greater than zero implies that the transition to the sticking mode is not possible. Similarly in the opposite direction, by the end of the sticking mode, the relative velocity is still equal to zero and a transition to slipping mode is not possible. A solution to these problems is to update $\dot{q}$ and ω such that the required condition on Δv^+ is satisfied.

Since equalities cannot be detected exactly during simulation, in the implementation of the simulation models the equality conditions have to be replaced with inequality conditions on the residual of the errors. We consider a constant $\epsilon > 0$ defining the numerical tolerance, e.g., $\Delta v = 0$ is implemented as $\|\Delta v\| \leq \epsilon$. Note that for numerical reasons, we also allow $h(q)$ to drift in the in-contact modes.

3.2 Hybrid model

The transitions between the modes of (44)–(47c) happen according to the discussion of Sect. 3.1. To avoid extra complexity, the contact types (point and sphere or line and sphere) are not distinguished by additional modes. The final model is a hybrid, variable-index DAE system. Figure 3 illustrates the hybrid model. In the figure, post-impact relative velocities are used in the conditions for the transition from the free motion mode. This is, however, just for saving space. It can be assumed that (50) is the transition condition to an "In-contact" state where the impact law is applied. Consequently, depending on $\|\Delta^+\|$ the transition to either sticking or slipping happens.

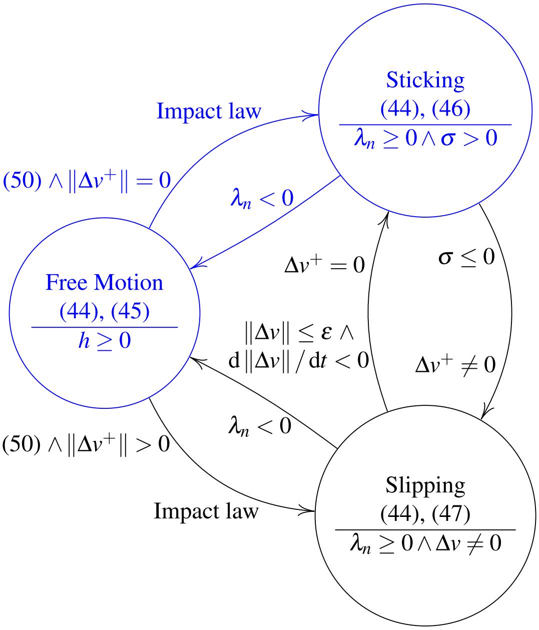

Sticking
(44), (46)
$\lambda_n \geq 0 \wedge \sigma > 0$
Impact law
(50) $\wedge \|\Delta v^+\| = 0$
$\lambda_n < 0$
Free Motion
(44), (45)
$h \geq 0$
$\Delta v^+ = 0$
$\sigma \leq 0$
$\|\Delta v\| \leq \varepsilon \wedge$
$\mathrm{d}\|\Delta v\| / \mathrm{d}t < 0$
$\Delta v^+ \neq 0$
$\lambda_n < 0$
(50) $\wedge \|\Delta v^+\| > 0$
Impact law
Slipping
(44), (47)
$\lambda_n \geq 0 \wedge \Delta v \neq 0$

Fig. 3 The hybrid system representing the ball and finger system. Only the subsystem shown in *blue* is used for finding optimal trajectories, limiting the possible solutions to free motion and sticking (Color figure online)

4 Optimization

In this section, we discuss how the hybrid DAE in Fig. 3 can be used to find trajectories for the system using optimal control. In the optimization, the slipping mode and its transitions are not considered, i.e., we only consider free motion and sticking, which correspond to the parts in blue in Fig. 3. For many applications, this choice is motivated since slippage implies energy loss, which is preferably avoided. However, this hybrid DAE is ill-suited for straightforward application, because of the combination of nonlinear and hybrid dynamics. To deal with the hybrid dynamics, we use a mixture of multiple phases and smooth approximations to obtain a system model that is better suited for numerical optimization. The resulting equations are nonlinear and sufficiently differentiable to apply Newton-based methods.

4.1 Optimal control problem

To investigate optimal trajectories, we use the optimal control problem formulation

$$\begin{aligned}
&\text{minimize} && \phi\big(c(t_f)\big), && \text{(51a)}\\
&\text{with respect to} && x, y, u, t_f, x(0), && \\
&\text{subject to} && \text{dynamics of Fig. 3 in blue}, && \text{(51b)}\\
& && \dot{c} = L(x, u), && \text{(51c)}\\
& && \gamma\big(x(0), x(t_f)\big) = 0, \quad c(0) = 0, && \text{(51d)}\\
& && g(\dot{x}, x, y, u) \leq 0, && \text{(51e)}
\end{aligned}$$

where

$$x = (q, dq, \omega), \qquad y = (h, \lambda, \lambda_n, \sigma), \qquad u = \tau \tag{52}$$

are the differential, algebraic, and control variables, respectively, t_f denotes the final time, c is additional state variables that model the objective defined by ϕ and L, which we describe further in Sect. 4.4, g provides path inequality constraints, which are described in Sect. 4.2, and γ are point constraints used to enforce, for example, initial and periodicity conditions. The initial state $x(0)$ can also be a (partially) free optimization variable.

This general formulation can be used to, e.g., find optimal solutions to the following problems:

- Reach the ball;
- Reorient the ball;
- Rotate the ball with a reference velocity;
- Rotate the ball along an axis as fast as possible;
- Find a periodic solution to keep the ball rotating.

4.2 From hybrid dynamics to multiple phases

The optimization problem (51a)–(51e) is not compatible with conventional methods for numerically solving optimal control problems as a result of the hybrid dynamics. To accurately treat the switching dynamics discussed in Sect. 3 in a differentiable way without making smooth approximations—which would lead to unacceptable inaccuracy—we introduce multiple phases as described in, e.g., [2, 6]. The time horizon $[0, t_f]$ is thus divided into N phases

$$\left[t_0^{(1)}, t_f^{(1)}\right], \quad \ldots, \quad \left[t_0^{(N)}, t_f^{(N)}\right] \tag{53}$$

such that

$$t_0^{(1)} = 0, \qquad t_f^{(N)} = t_f, \qquad t_0^{(i+1)} = t_f^{(i)}, \quad i = 1, \ldots, N-1. \tag{54}$$

Within each phase only a single, prescribed mode is active, and the mode is only allowed to switch at the phase boundaries.

For ease of notation, let

$$z^{(i)} := \begin{bmatrix} \dot{x}^{(i)} & x^{(i)} & \dot{c}^{(i)} & c^{(i)} & y^{(i)} & u^{(i)} & t \end{bmatrix}^T. \tag{55}$$

The dynamics within phase i are then described by the non-hybrid DAE

$$F^{(i)}\left(z^{(i)}\right) = 0 \tag{56}$$

that corresponds to the mode that has been prescribed to the phase, that is, (45) or (46), as well as (44) and (51c).

The validity conditions are enforced by the path constraint $g^{(i)}$. State and control inequality constraints can also be incorporated into $g^{(i)}$. Event functions $e^{(i)}$ are used to enforce mode transition conditions. The system variables in the respective phases are connected by linkage constraints ψ corresponding to the reset maps of Fig. 3.

The hybrid dynamic optimization problem (51a)–(51e) is thus transformed into the non-hybrid, multiple phase problem

$$
\begin{aligned}
&\text{minimize} && \phi\bigl(c^{(N)}(t_f)\bigr), && \text{(57a)}\\
&\text{w.r.t.} && x^{(i)}, c^{(i)}, y^{(i)}, u^{(i)}, t_f^{(i)}, x_0^{(i)}, &&\\
&\text{subject to} && F^{(i)}\bigl(z^{(i)}\bigr) = 0, && \text{(57b)}\\
& && e_L^{(i)} \le e^{(i)}\bigl(z^{(i)}\bigl(t_0^{(i)}\bigr), z^{(i)}\bigl(t_f^{(i)}\bigr)\bigr) \le e_U^{(i)}, && \text{(57c)}\\
& && \psi_L \le \psi\bigl(z^{(1)}\bigl(t_0^{(1)}\bigr), z^{(1)}\bigl(t_f^{(1)}\bigr), &&\\
& && \qquad\qquad \vdots &&\\
& && \qquad z^{(N)}\bigl(t_0^{(N)}\bigr), z^{(N)}\bigl(t_f^{(N)}\bigr)\bigr) \le \psi_U, && \text{(57d)}\\
& && g^{(i)}\bigl(z^{(i)}\bigr) \le 0, && \text{(57e)}\\
& && i = 1, \ldots, N, &&
\end{aligned}
$$

where $e_L^{(i)}$, $e_U^{(i)}$, ψ_L, ψ_U are proper constants. Any remaining point constraints given by (51d) have also been incorporated into either the event constraints (57c) or the linkage constraints (57d).

4.3 Smooth saturation

Although (57a)–(57e) is not hybrid, it is still not suitable for dynamic optimization because of the non-differentiable equations resulting from the saturation function in (3). While this can be handled by introducing further phase changes for whenever a junction point of the saturation function is crossed—i.e., when the potential contact point shifts between the fingertip to the side of the finger—this approach is computationally expensive and also makes it more difficult to determine the optimal phase sequence. We thus instead choose to approximate the saturation function by noting that

$$\operatorname{sat}_L^U(x) = \frac{1}{2}\bigl(L + U + |x - L| - |x - U|\bigr) \tag{58}$$

and then using the approximation

$$|x| \approx \sqrt{x^2 + \epsilon^2}, \tag{59}$$

where ϵ is a small number, which we set equal to 0.01. Figure 4 compares the smooth approximation with the ideal saturation, as well as the hyperbolic tangent function with an appropriate shifting and scaling. The hyperbolic tangent is another function commonly used to model saturation behavior [25]. Note that a significantly better approximation by scaling the hyperbolic tangent function is not possible since it requires changing either the slope or the saturation limits.

4.4 Objective

Out of the various archetype problems mentioned in Sect. 4.1, we focus on the final one of finding a periodic trajectory, viz., $x(0) = x(t_f)$, that achieves a certain rotation of the ball.

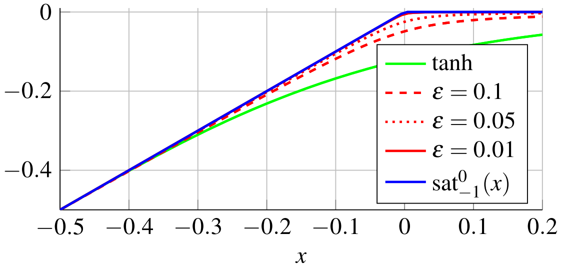


Fig. 4 Approximation of the saturation function with the lower limit of -1 and the upper limit of 0. The curves have point symmetry around $(-0.5, -0.5)$. Hence, only the first quadrant is shown. The *green curve* approximates the saturation function by scaling and shifting tanh. We observe that for $\epsilon = 0.01$, the smooth approximation of the saturation function is hardly distinguishable from the saturation function (Color figure online)

We use the objective function

$$\phi\big(c(t_f)\big) := \frac{c_1(t_f)}{c_2(t_f)}, \tag{60}$$

where c_1 and c_2 denote some cost states. Accordingly, $\phi(c(t_f))$ is the ratio between the total accumulated costs during the time horizon $[0, t_f]$. We will consider two different objective functions, hence two different ways of defining c_1 and c_2. The first is cost of transport (COT), which can be expressed by the two cost states

$$\dot{c}_1^{\mathrm{COT}} := \|u\|^2, \tag{61}$$

which quantifies the cost of control action, and a measure of the amount of rotation in a desired direction

$$\dot{c}_2^{\mathrm{COT}} := -w^{\mathrm{COT}} \big\|\underline{\omega}^{\mathrm{ref}}\big\|^2 + \omega \cdot \hat{\omega}^{\mathrm{ref}}, \tag{62}$$

where $\hat{\omega}^{\mathrm{ref}}$ is a unit vector determining the desired direction of angular velocity,

$$\underline{\omega}^{\mathrm{ref}} := \omega - \big(\omega \cdot \hat{\omega}^{\mathrm{ref}}\big)\hat{\omega}^{\mathrm{ref}} \tag{63}$$

is the component of the angular velocity perpendicular to the desired direction, and w^{COT} is used to weight the two quantities. The second objective is a more conventional quadratic penalty on the input weighted by w^{QC} and on the deviation from a reference rotational velocity, ω^{ref}, expressed by the cost states

$$\dot{c}_1^{\mathrm{QC}} := w^{\mathrm{QC}}\|u\|^2 + \big\|\omega - \omega^{\mathrm{ref}}\big\|^2, \tag{64}$$

and a state for calculating the total period

$$\dot{c}_2^{\mathrm{QC}} := 1. \tag{65}$$

The cost of transport defined here is motivated by the energy efficiency of the rotation and is similar to the energy index used for comparing the efficiency of various transport systems [31]. The average rotational speed is thus determined by the optimization. On the other hand, the quadratic objective function is intended for achieving a certain average rotational speed while allowing some trade-off for the average power. The fact that the quadratic cost function favors angular velocities close to the constant reference velocity leads to short periods. This is understood by noting that the speed of the ball cannot be influenced in the free motion and drops rapidly.

5 Implementation

The equations of Appendices A and B were derived with the use of Maple. The multiphase problem (57a)–(57e) was implemented in PSOPT [3]. The Legendre–Gauss pseudospectral collocation of PSOPT was used to transcribe the problem into a twice continuously differentiable nonlinear program (NLP) with 30 collocation points for each phase. The NLP was solved with IPOPT [34] and the linear solver MA27 [18].

Simulation of the system including slipping was performed using Modelica [14] and Dymola [11]. Dymola does not currently support hybrid systems with variable index, although there has been recent work to resolve this [24]. One possible way to manually resolve this is to perform index reduction in the high-index modes, but this is challenging for the considered system because of the impossibility of statically selecting suitable state variables. Hence, we instead adopted a more cumbersome approach where three instances of the system were simulated simultaneously, one low-index for free motion and two high-index (with dynamic state selection [23]) for modes in contact. A correct system model could thus be created by appropriate communication between the three instances during mode switching.

The trajectories from both optimization and simulation were finally imported into MATLAB for visualization. For 3D visualization of the finger and its movements, Peter Corke's robotic toolbox [9] was used.

6 Results

This section presents results for the problem of finding a periodic solution for rotating the ball. We first present optimal trajectories obtained by neglecting slipping, and then briefly show the solutions obtained by simulating the system with the open-loop optimal inputs while taking into account slippage. For the numerical experiments, we use $r = \ell = 1$ m, $m = 20$ g, and $I_b = \operatorname{diag}(1.0677, 2.3177, 2.0833)$ g s^{-2} for all the links.

6.1 Optimal control

We considered the two objectives discussed in Sect. 4.4. We divided each period into three phases: free motion, sticking, and finally free motion again with the constraint $x^{(3)}(t_f) = x^{(1)}(0)$. The initial state as well as the terminal and switching times $t_f^{(i)}$ were free. We also imposed the initial constraint $h(0) \geq 0.3r$ for robustness, i.e., the finger should go sufficiently far away from the ball before making a new contact. The weights w^{COT} and w^{QC} were optionally set to 10 and 1, respectively. As the desired direction of angular velocity $\hat{\omega}^{\mathrm{ref}}$, we chose $1/\sqrt{2}(1, -1, 0)^T$ for the cost of transport and we used $\omega^{\mathrm{ref}} = (0, 0, 2.2)^T$ as the angular velocity reference for the quadratic objective. We impose an arbitrary torque limit of 1 N m.

A simple initial guess was sufficient to numerically solve the problem. The initial guess consisted of a constant non-zero $q \equiv q^0$ and $t_f^{(i)} = i$, with all the remaining variables being 0.

We present the results of three scenarios. In the first two scenarios, we solved the problem of optimizing the COT. The difference between the scenarios is that in the first one we

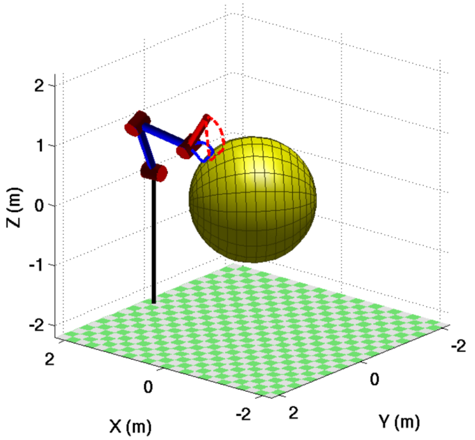


Fig. 5 Visualization of the ball-and-finger system. The *blue solid curve* shows the optimal path of p_c for the COT when there is no joint limit. The *red dashed curve* corresponds the optimal path for the fingertip (Color figure online)

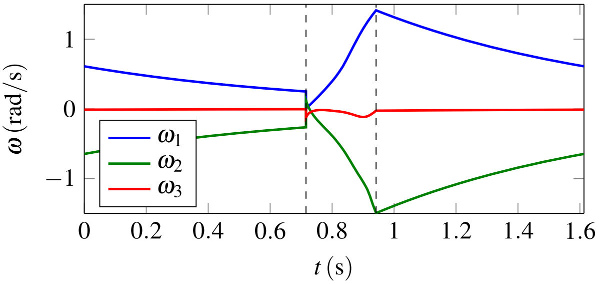


Fig. 6 The optimal angular velocity of the ball ω for the COT. The *vertical lines* show the switching time for the phases

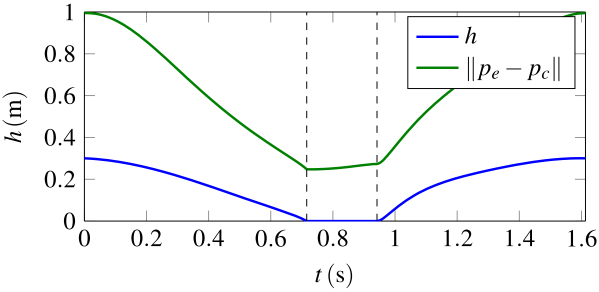


Fig. 7 The minimum distance to the ball, h, and the distance of the contact point to the fingertip. The finger is in contact with the ball in the interval between the *vertical lines*. The side of the finger is used for rotating the ball and the contact point shifts slightly during rotation

included no constraint on the joint angles, while in the second one the constraint $q_i \geq 0$ for $i \in \{2, 3\}$ was enforced in order to comply with the limitations of the human finger. Figure 5 illustrates the optimal path obtained for the robotic finger. In Fig. 6, we see the resulting optimal angular velocities of the ball and switching times. The discontinuities at the impact point are visible.

The minimum distance h between the finger and the ball is shown in Fig. 7. Moreover, the distance between the contact point and the fingertip is plotted. As we see, the contact point is located somewhere on the link and shifts slightly during the rotation. The required torques and the interaction force are given in Figs. 8 and 9, respectively.

Figure 10 illustrates the human finger in its initial position. In this case, the fingertip always remains the closest point to ball. The path obtained from the movement of the closest

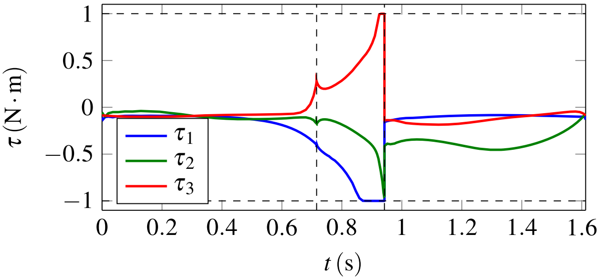


Fig. 8 The optimal control signal for minimizing the COT: The *vertical lines* show the switching time for the phases. Toward the end of the contact phase, the bounds on the control signals become active

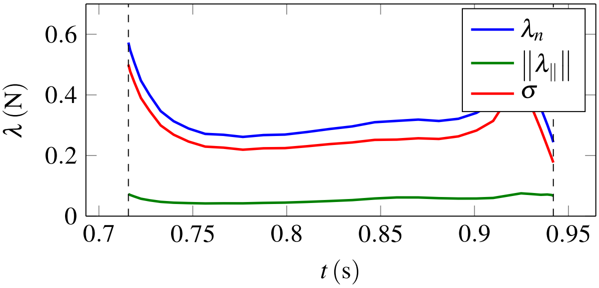


Fig. 9 Interaction forces for the COT: The tangential and normal components of the interaction force are shown. In this case, there is always some margin to slipping

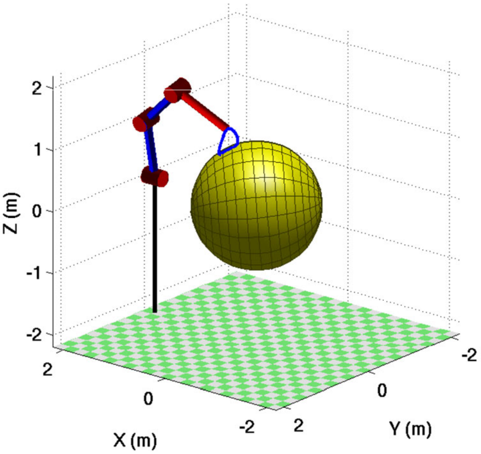


Fig. 10 Visualization of the ball and finger system. The *blue curve* shows the optimal path of p_c for the COT, complying with the human joint limits. The contact point is always the fingertip (Color figure online)

point is shown in blue. The paths of the closest points of the human finger and the robotic finger are compared in Fig. 11.

Assuming the COT, the robotic finger was able rotate the ball in the desired direction more efficiently. It used the side of the finger, while the human finger contacted with the fingertip only.

Figure 12 shows the finger in its initial position and the path obtained for rotating the ball around the z-axis using the quadratic objective function. The angular velocity obtained is illustrated in Fig. 13. As we can see, the average angular velocity matches the desired ω_{ref}. The period of the solution compared to Fig. 6 is shorter. The minimum distance as a function of time and the distance of the contact point to the fingertip are shown in Fig. 14. In this solution, only the fingertip is used for rotating the ball. Finally, the torques and interaction

Fig. 11 The paths of p_c obtained using the COT optimization criterion. The *red* and *blue curves* correspond to human and robotic fingers, respectively. The parts of the paths in contact with the ball are *dashed* (Color figure online)

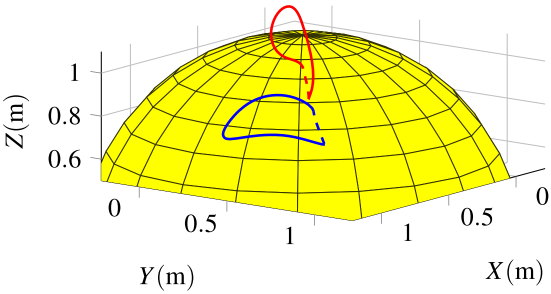


Fig. 12 Visualization of the ball and finger system. The *blue curve* shows the optimal path of p_c for the quadratic cost. The objective is to rotate the ball around the z-axis (Color figure online)

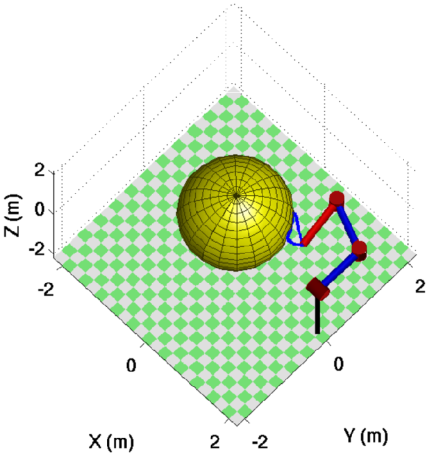
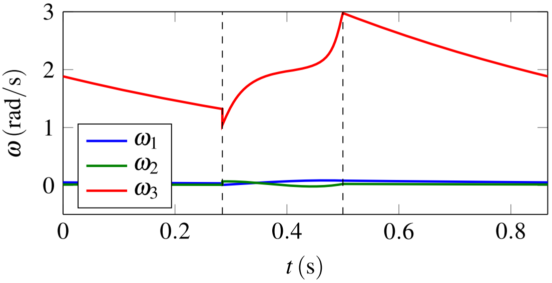


Fig. 13 The optimal angular velocity of the ball ω for the quadratic cost. The *vertical lines* show the switching time for the phases



forces are depicted in Figs. 15 and 16, respectively. As we see, the constraint on the friction cone is always active during contact. Hence, the motion is always on the verge of slipping.

6.2 Open-loop simulation

We now use the open-loop optimal torques for the COT criterion without non-negativity bounds on the joints from Fig. 8 to simulate the full system in Fig. 3 in Dymola. The resulting angular velocities are shown in Fig. 17 and compared with the optimal angular velocities. We see that open-loop angular velocities match well at the beginning, but drift from the optimum as time passes. There is a short window of time immediately following collision in

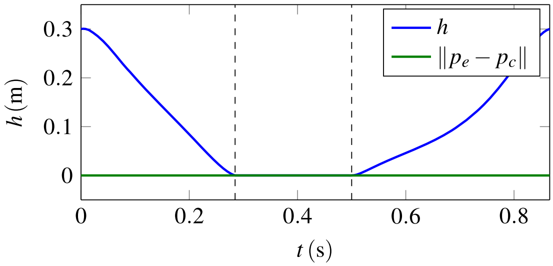


Fig. 14 The minimum distance to the ball, h, and the distance of the contact point to the fingertip. The finger is in contact with the ball in the interval between the *vertical lines*. Only the fingertip is used for rotating the ball

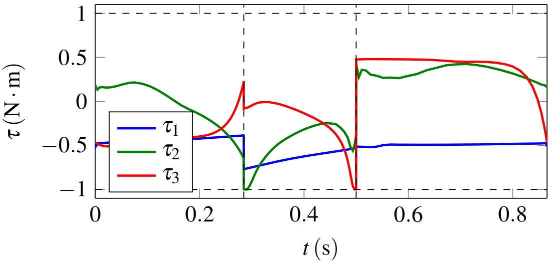


Fig. 15 The optimal control signal for minimizing the quadratic cost: The *vertical lines* show the switching time for the phases. Note that the bounds on the control signal are only active when making contact or breaking it

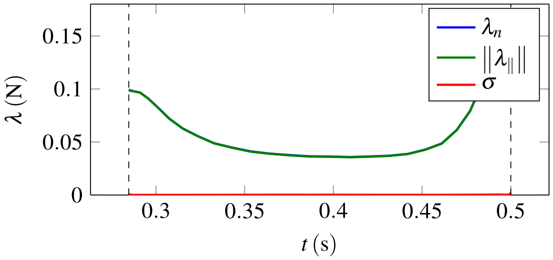


Fig. 16 Interaction forces for the quadratic cost: The tangential and normal components of the interaction force are shown. In this case, there is no margin to slipping

which slipping occurs, which is almost always the case when employing Newton's impact law. The dynamics quickly switch to sticking because of the high interaction forces. There is a jump in the post-impact velocities in the optimization solution, while using the Newton's impact law, the velocities are continuous in the simulation result.

7 Discussion

The ball-and-finger system is modeled with three modes, each described with a different set of equations. In contrast to some of the results for finding optimal trajectories reported in the literature (see, for example, [20, 30]), in our approach the connection between different modes happens through phases. Thus, it is not required to compromise the optimal solution by biasing the cost function toward making contact. On the other hand, we have fixed the order of the phases. This can be relaxed by trying all the combinations. For more complex systems, a hierarchical planning scheme can be used in order to determine the phases [15].

There are several possible approaches to treating hybrid dynamics in dynamic optimization other than the multiphase framework. A natural alternative in the considered case is the incorporation of the complementarity constraints [1] in the optimization problem. However, it is not always easy to rewrite the non-smooth behavior using complementarity conditions,

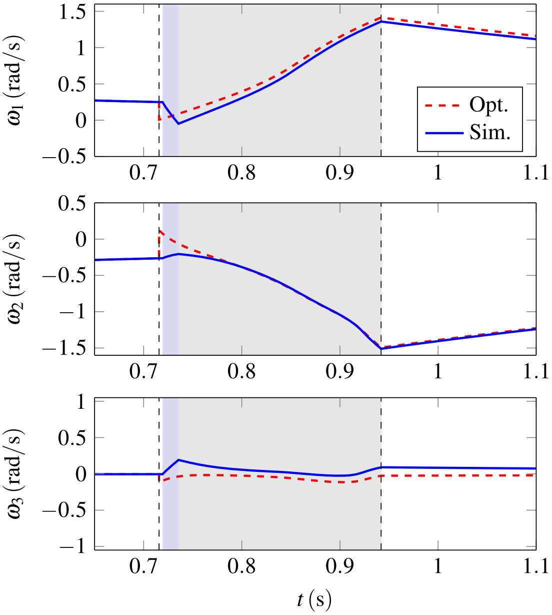


Fig. 17 The open-loop simulation of the complete system in Fig. 3 using the control signals in Fig. 8. The angular velocities of the ball (*solid blue curves*) are compared with the optimization result (*red dashed curves*). The *vertical lines* show the switching times obtained from the optimization. The time regions corresponding to the slipping and sticking modes of the simulated result are shaded in *purple and gray*, respectively. We see that the contact times are slightly different. However, there is no difference in the time that the finger leaves the ball (Color figure online)

e.g., switching between slipping and sticking when the friction coefficients are not the same. Other possible approaches are mixed-integer nonlinear programming based on branch-and-bound [4] or sum-up-rounding [19] or widely applicable derivative-free methods such as genetic algorithms [33]. The various approaches have their own strengths and weaknesses. The strength of the multiphase framework is the efficiency of the required computations, which may enable online solution of these problems. A significant drawback is that the phase sequence needs to be determined a priori.

In optimization, it is not always suitable to neglect the slippage completely. It can be ignored when the interaction forces are high. However, when the slipping margin σ is small, it becomes more important to consider the effects of slipping. If the slipping mode is neglected, it is also important to ensure that the collision itself does not give rise to slipping. When using Newton's impact law, this is possible by constraining the relative tangential velocity at the collision point to be zero. This will guarantee a transition to the sticking mode upon collision, though it might not be the optimal solution. Another solution is to constrain the resulting impulse ι so that the friction is sufficient to support the sticking mode. Similar to (19), this can be formulated as

$$\|\iota_{\|}\| \leq \mu_s \iota \cdot n, \tag{66}$$

and enforced in the optimization using event constraints. In our example with the quadratic objective function in Sect. 6.1, adding this constraint did not have a significant effect on the solution. For the COT objective function, the velocity of the finger normal to the ball was prior to the collision increased to ensure enough friction while the rest of the variables were similar. This suggests that introducing limits on the normal impulse might also be required to avoid bouncing back in practice.

Another option for handling impacts is to add flexible components such as a non-rigid ball or a padded arm. This leads to a smooth approximation of discontinuities, which does not require the exact contact times to be identified. For example, the normal force can be considered proportional to the depth of penetration of p_c, i.e., $\lambda_n = -k \min(h(q), 0)n$ could be employed. This is, however, not suitable when the contact stiffness is high, since a computationally intractable amount of discretization points may be required to be able to capture rapid changes in the forces. Note that, irrespective of this assumption, it is possible to constrain the amount of normal impulse in the optimization such that the desired behavior is achieved.

Contrary to assumption 6, real contacts have some degree of elasticity. However, given the fact that both robotic and human fingers have relatively high impedance, the normal impact at the collision time is absorbed and the finger would not bounce back. Therefore, an inelastic model is able to describe the overall behavior well for practical collision velocities.

Open-loop control is generally not satisfactory, even if slippage can be neglected. Feedback is needed to handle model uncertainties, such as an inaccurate collision model, as well as discretization errors. Different controllers depending on the mode of the system may be employed. Specifically, in free motion, a position controller with feedforward torques can be used. As soon as the finger comes into contact with the ball, the controller can be switched to a hybrid position/force controller. To maintain the sticking condition, the normal force should be regulated while the velocity in the tangential direction is being tracked.

In simulation, inconsistent solutions due to the Painlevé paradox [8] may arise when the kinetic friction constant between the ball and the finger is large. In this case, without a proper reinitialization of the states, it is not possible to find consistent solutions with the complementarity conditions (48). However, this will not be an issue in the optimization because of the infeasibility of such solutions.

The optimization results presented in Sect. 6 were obtained using simple initial guesses. However, warm starting the optimization from previous runs with simplified cost functions can often be beneficial. This way it is possible to direct the solution gradually to the region of interest.

The path constraint related to the validity condition $h \geq 0$ is difficult to treat numerically. In phase 1, the natural formulation is $h^{(1)} \geq 0$ and $h^{(1)}(t_f) = 0$. However, such a formulation violates constraint qualifications, leading to unbounded dual variables [29]. Ideally, this would be handled by only enforcing $h^{(1)} \geq 0$ in all but the last collocation point of phase 1, which instead is determined by the event constraint $h^{(1)}(t_f^{(1)}) = 0$, but this is not readily supported by PSOPT. Hence, we work around this issue by replacing the event constraint by $0 \leq h^{(1)}(t_f) \leq \epsilon := 10^{-4}$. Further problems related to this are encountered in phase 3. In phase 2, $\dot{h} = 0$ is a solution invariant, which is, however, not preserved during discretization. Numerical drift-off can thus lead to $h^{(2)}(t_f) < 0$ (enforcing $h^{(2)} \geq 0$ would once again lead to violation of constraint qualifications). To circumvent this, we replace the validity condition of phase 3 by $h^{(3)} \geq -\epsilon$.

The proposed framework is not robust with respect to avoiding contact in the free motion mode, as it allows for free motion arbitrarily close to the ball. A simple remedy for the considered examples is to also enforce $\dot{h} \leq -\epsilon$ in the first phase and $\dot{h} \geq \epsilon$ in the third phase, which may, however, sacrifice optimality. Future work is to design a more sophisticated solution by considering robust approaches to optimization and control such as those in [5, 27].

8 Conclusion

In this article, we propose a ball-and-finger system as a realistic example to study common robotic tasks involving interaction and varying dynamics. The result of modeling is a hybrid, variable-index differential-algebraic equation system. To find the optimal trajectories for performing a task, we used an integrated approach, i.e., there is no separate process for finding the required path. A multiphase formulation was used to handle changes in the dynamics in each mode. The benefit of this approach is that there is no need to compromise the desired objective function in order to establish contact.

As the result of the optimization, we find not only the trajectory for the motion but also the interaction forces. This is an important feature considering the fact that robots are often employed for manipulation tasks, which involves interaction with other objects. The extension of our model with several fingers opens up the opportunity to perform automatic motion planning for dexterous manipulation, e.g., for assembly tasks.

From the modeling, three modes of behavior from the system are expected, which may require three different strategies for control. Suitable feedback signals are the minimum distance to the ball, the margin to slipping, and the relative velocity at the contact point. In the contact situation, controlling the force in the normal direction to the ball is important in order to maintain the contact as well as to make the system more robust with respect to the variations of the impact law.

Acknowledgements The research has received funding from the LCCC Linnaeus Center at Lund University and the SOMA project (European Commission, Horizon 2020 Framework Programme, H2020-ICT-645599).

Appendix A: Forward kinematics and Jacobian

Considering Fig. 1, the position of the fingertip, p_e, and the orientation of the 3rd link, described as a rotation matrix, are

$$p_e = \begin{bmatrix} -s_{23} - s_2 + 2 \\ \frac{1}{2}(s_{123} + s_{1\bar{2}\bar{3}} + s_{12} + s_{1\bar{2}}) + s_1 \\ \frac{1}{2}(c_{1\bar{2}\bar{3}} + c_{123} + c_{1\bar{2}} + c_{12}) + c_1 \end{bmatrix} \ell, \tag{67}$$

$$[\hat{e}_{x_3}\ \hat{e}_{y_3}\ \hat{e}_{z_3}] = \begin{bmatrix} -s_{23} & -c_{23} & 0 \\ \frac{1}{2}(s_{123} + s_{1\bar{2}\bar{3}}) & \frac{1}{2}(c_{123} - c_{1\bar{2}\bar{3}}) & -c_1 \\ \frac{1}{2}(c_{1\bar{2}\bar{3}} + c_{123}) & \frac{1}{2}(s_{1\bar{2}\bar{3}} - s_{123}) & s_1 \end{bmatrix}, \tag{68}$$

where $s_{ij\ldots k}$ denotes $\sin(q_i + q_j + \cdots + q_k)$ and $c_{ij\ldots k}$ denotes $\cos(q_i + q_j + \cdots + q_k)$. An index with a bar denotes the corresponding joint with a negative sign, e.g., $\bar{i}$ represents $-q_i$.

The geometric Jacobian for the fingertip is

$$J = \begin{bmatrix} J_p \\ J_o \end{bmatrix}, \tag{69}$$

where the translational part, J_p, is given by

$$J_p = \begin{bmatrix} 0 & -c_{23} - c_2 & -c_{23} \\ J_{21} & \frac{1}{2}(c_{123} - c_{1\bar{2}\bar{3}} - c_{1\bar{2}} + c_{12}) & \frac{1}{2}(c_{123} - c_{1\bar{2}\bar{3}}) \\ J_{31} & \frac{1}{2}(s_{1\bar{2}\bar{3}} - s_{123} - s_{12} + s_{1\bar{2}}) & -\frac{1}{2}(s_{1\bar{2}\bar{3}} - s_{123}) \end{bmatrix} \ell, \tag{70}$$

where

$$J_{21} = \frac{1}{2}(c_{1\bar{2}\bar{3}} + c_{123} + c_{1\bar{2}} + c_{12}) + c_1,$$

$$J_{31} = -\frac{1}{2}(s_{123} + s_{1\bar{2}\bar{3}} + s_{12} + s_{1\bar{2}}) - s_1,$$

and the rotational part, J_o, is given by

$$J_o = \begin{bmatrix} -1 & 0 & 0 \\ 0 & -c_1 & -c_1 \\ 0 & s_1 & s_1 \end{bmatrix}. \tag{71}$$

Appendix B: Dynamics

In this section, the expressions for the dynamical properties of the finger used in (24) are given. Every link is characterized by the mechanical properties mass m_i and moments of inertia $I_i = \operatorname{diag}(I_{xx_i}, I_{yy_i}, I_{zz_i})$ where i denotes the link number.

The expression for the gravitational force is

$$g(q) = -\frac{\ell}{4} g_0 \begin{bmatrix} g_{11} \\ m_2(s_{11} - s_{1\bar{1}}) + (s_{111} - s_{1\bar{1}\bar{1}} + 2s_{11} - 2s_{1\bar{1}})m_3 \\ m_3(s_{111} - s_{1\bar{1}\bar{1}}) \end{bmatrix}, \tag{72}$$

$$\begin{aligned} g_{11} &= 2m_1 s_1 + (s_{11} + s_{1\bar{1}} + 4s_1)m_2 \\ &\quad + (s_{111} + s_{1\bar{1}\bar{1}} + 2s_{11} + 2s_{1\bar{1}} + 4s_1)m_3. \end{aligned} \tag{73}$$

Here s_{xyz} denotes $\sin(xq_1 + yq_2 + zq_3)$ and c_{xyz} denotes $\cos(xq_1 + yq_2 + zq_3)$. A variable with a bar denotes the variable with a negative sign, e.g., $\bar{x}$ represents $-x$. The trailing zeros are omitted.

The mass matrix for the finger is

$$\begin{aligned} M_f(q) &= M_f^T(q) \\ &= \begin{bmatrix} m_{11} & 0 & 0 \\ 0 & \frac{1}{4}m_2\ell^2 + (c_{001} + \frac{5}{4})m_3\ell^2 + I_{zz_2} + I_{zz_3} & \frac{1}{4}(2c_{001} + 1)m_3\ell^2 + I_{zz_3} \\ 0 & \frac{1}{4}(2c_{001} + 1)m_3\ell^2 + I_{zz_3} & \frac{1}{4}m_3\ell^2 + I_{zz_3} \end{bmatrix}, \end{aligned} \tag{74}$$

where

$$\begin{aligned} m_{11} &= \frac{1}{8}\big(-4I_{xx_3} + 4I_{yy_3} + m_3\ell^2\big)c_{022} + \frac{1}{2}m_3\ell^2 c_{021} + \frac{1}{8}(8c_{01} + 9)m_2\ell^2 \\ &\quad + \frac{1}{8}\big((m_2 + 4m_3)\ell^2 - 4I_{xx_2} + 4I_{yy_2}\big)c_{020} + \frac{1}{2}(I_{yy_2} + I_{xx_3} + I_{yy_3} + I_{xx_2}) \\ &\quad + \frac{1}{8}(16c_{01} + 4c_{001} + 8c_{011} + 13)m_3\ell^2 + I_{yy_1} + \frac{1}{4}m_1\ell^2. \end{aligned} \tag{75}$$

The matrix $C(q,\dot{q})$ can be expressed as

$$C_{ij} = \sum_{k=1}^{3} \psi_{ijk}\dot{q}_k, \tag{76}$$

where $\psi_{ijk} = 0$ except for

$$\begin{aligned}\psi_{112} = \psi_{121} = -\psi_{211} &= \frac{1}{8}\big(4I_{xx_3} - 4I_{yy_3} - m_3\ell^2\big)s_{022} - \frac{1}{2}m_3\ell^2 s_{021} \\ &\quad - \frac{1}{8}\big((4m_3 + m_2)\ell^2 - 4I_{xx_2} + 4I_{yy_2}\big)s_{02} - \frac{1}{2}m_3\ell^2 s_{011} \\ &\quad - \left(\frac{1}{2}m_2 + m_3\right)\ell^2 s_{01},\end{aligned} \tag{77}$$

$$\begin{aligned}\psi_{113} = \psi_{131} = -\psi_{311} &= \frac{1}{8}\big(4I_{xx_3} - 4I_{yy_3} - m_3\ell^2\big)s_{022} \\ &\quad - \frac{1}{4}m_3\ell^2(2s_{011} + s_{001} + s_{021}),\end{aligned} \tag{78}$$

$$\psi_{223} = \psi_{232} = \psi_{233} = -\psi_{322} = -\frac{1}{2}m_3\ell^2 s_{001}. \tag{79}$$

References

1. Baumrucker, B.T., Biegler, L.T.: MPEC strategies for optimization of a class of hybrid dynamic systems. J. Process Control **19**(8), 1248–1256 (2009)
2. Becerra, V.: PSOPT optimal control solver user manual (release 3) (2010). Available at: http://code.google.com/p/psopt/downloads/list
3. Becerra, V.M.: Solving complex optimal control problems at no cost with PSOPT. In: Proc. IEEE Multi-Conf. Systems and Control, Sep. 7–10, Yokohama, Japan, pp. 1391–1396 (2010)
4. Belotti, P., Lee, J., Liberti, L., Margot, F., Wächter, A.: Branching and bounds tightening techniques for non-convex MINLP. Optim. Methods Softw. **24**(4–5), 597–634 (2009)
5. Ben-Tal, A., El Ghaoui, L., Nemirovski, A.: Robust Optimization. Princeton University Press, Princeton (2009)
6. Betts, J.T.: Practical Methods for Optimal Control and Estimation Using Nonlinear Programming, 2nd edn. SIAM, Philadelphia (2010)
7. Bloch, A.M.: Nonholonomic Mechanics and Control. Springer, New York (2003)
8. Brogliato, B.: Nonsmooth Mechanics: Models, Dynamics and Control, 3rd edn. Springer, Berlin (2012)
9. Corke, P.I.: Robotics, Vision & Control: Fundamental Algorithms in Matlab. Springer, Berlin (2011)
10. Denavit, J., Hartenberg, R.S.: A kinematic notation for lower-pair mechanisms based on matrices. J. Appl. Mech. **22**, 215–221 (1955)
11. Dymola: Multi-Engineering Modeling and Simulation—Dymola—CATIA (2016). http://www.dymola.com
12. Felis, M.L., Mombaur, K.: Synthesis of full-body 3-D human gait using optimal control methods. In: Proc. Int. Conf. Rob. Auto. (ICRA), Stockholm, Sweden, pp. 1560–1566 (2016)
13. Fouquet, M., Guéguen, H., Faille, D., Dumur, D.: Hybrid dynamic optimization of power plants using sum-up rounding and adaptive mesh refinement. In: Proc. IEEE Conf. on Control Applications, Antibes, France, pp. 316–321 (2014)
14. Fritzson, P.: Principles of Object-Oriented Modeling and Simulation with Modelica 3.3, 2nd edn. Wiley/IEEE Press, Piscataway (2015)
15. Ojer de Andrés, M., Mahdi Ghazaei Ardakani, M., Robertsson, A.: Reinforcement Learning for 4-Finger-Gripper Manipulation. Accepted for publication in Proc. IEEE Int. Conf. Rob. Aut. (ICRA), Brisbane, Australia (2018)
16. Goyal, S.: Planar sliding of a rigid body with dry friction: limit surfaces and dynamics of motion. Ph.D. Thesis, Cornell University (1989)

17. Hardt, M., von Stryk, O.: Dynamic modeling in the simulation, optimization, and control of bipedal and quadrupedal robots. Z. Angew. Math. Mech. **83**(10), 648–662 (2003)
18. HSL: A collection of Fortran codes for large scale scientific computation (2016). Software available at: http://www.hsl.rl.ac.uk
19. Kirches, C., Lenders, F.: Approximation properties and tight bounds for constrained mixed-integer optimal control. Math. Program. (2016, submitted for publication). Preprint available at Optimization Online; April, 2016
20. Kumar, V., Tassa, Y., Erez, T., Todorov, E.: Real-time behaviour synthesis for dynamic hand-manipulation. In: Proc. IEEE Int. Conf. Rob. Auto. (ICRA), Hong Kong, China, pp. 6808–6815 (2014)
21. Kumar, V., Todorov, E., Levine, S.: Optimal control with learned local models: application to dexterous manipulation. In: Proc. IEEE Int. Conf. Rob. Auto. (ICRA), Stockholm, Sweden, pp. 378–383 (2016)
22. Lin, H., Antsaklis, P.J.: Hybrid dynamical systems: an introduction to control and verification. Found. Trends Syst. Control **1**(1), 1–172 (2014)
23. Mattsson, S.E., Olsson, H., Elmqvist, H.: Dynamic selection of states in Dymola. In: Proc. Modelica Workshop 2000, Lund, Sweden, pp. 61–67 (2000)
24. Mattsson, S.E., Otter, M., Elmqvist, H.: Multi-mode DAE systems with varying index. In: Proc. 11th Int. Modelica Conf, Paris, France, pp. 89–98 (2015)
25. Menon, A., Mehrotra, K., Mohan, C.K., Ranka, S.: Characterization of a class of sigmoid functions with applications to neural networks. Neural Netw. **9**(5), 819–835 (1996)
26. Meriam, J.L., Kraige, L.G.: Engineering Mechanics: Dynamics, vol. 2, 7th edn. Wiley, New Jersey (2012)
27. Mordatch, I., Lowrey, K., Todorov, E.: Ensemble-CIO: full-body dynamic motion planning that transfers to physical humanoids. In: Proc. IEEE/RSJ Int. Conf. Intell. Rob. Sys. (IROS), Hamburg, Germany, pp. 5307–5314 (2015)
28. Newton, I.: Mathematical Principles of Natural Philosophy. University of California Press, Berkeley and Los Angeles (1966). The original work 1687, English translation by A. Motte, 1729, translation revised by F. Cajori
29. Nocedal, J., Wright, S.: Numerical Optimization, 2nd edn. Springer, New York (2006)
30. Rombokas, E., Malhotra, M., Theodorou, E.A., Todorov, E., Matsuoka, Y.: Reinforcement learning and synergistic control of the ACT hand. IEEE/ASME Trans. Mechatron. **18**(2), 569–577 (2013)
31. Shi, W., Stapersma, D., Grimmelius, H.T.: Comparison study on moving and transportation performance of transportation modes. Int. J. Energy Environ. **2**(4), 106–120 (2008)
32. Tassa, Y., Erez, T., Todorov, E.: Synthesis and stabilization of complex behaviors through online trajectory optimization. In: Proc. IEEE/RSJ Int. Conf. Intell. Rob. Sys. (IROS), Vilamoura, Portugal pp. 4906–4913 (2012)
33. Thieriot, H., Nemura, M., Torabzadeh-Tari, M., Fritzson, P., Singh, R., Kocherry, J.J.: Towards design optimization with OpenModelica emphasizing parameter optimization with genetic algorithms. In: Proc. 8th Int. Modelica Conf, Dresden, Germany, pp. 756–762 (2011)
34. Wächter, A., Biegler, L.T.: On the implementation of a primal-dual interior point filter line search algorithm for large-scale nonlinear programming. Math. Program. **106**, 25–57 (2006)
35. Yamada, T., Yamada, M., Yamamoto, H.: Stability analysis of multiple objects grasped by multifingered hands with revolute joints in 2D. In: Proc. IEEE Int. Conf. Mechatronics and Automation (ICMA), Chengdu, China, pp. 1785–1792 (2012)
36. Yamada, T., Yamamoto, H.: Grasp stability analysis of multiple objects including contact surface geometry in 3D. In: Proc. IEEE Int. Conf. Mechatronics and Automation (ICMA), Takamatsu, Japan, pp. 36–43 (2013)