Mechanism and Machine Theory 69 (2013) 337–349

Contents lists available at ScienceDirect

Mechanism and Machine Theory

journal homepage: www.elsevier.com/locate/mechmt

# State constrained optimal control of a ball pitching robot

Esubalewe Lakie Yedeg *, Eddie Wadbro

*Department of Computing Science, Umeå University, SE-901 87 Umeå, Sweden*

ARTICLE INFO

*Article history:*
Received 30 January 2012
Received in revised form 13 June 2013
Accepted 21 June 2013
Available online 18 July 2013

*Keywords:*
Motion planning
Optimal control
Underactuated system
Adjoint method

ABSTRACT

We present a method for offline optimal control of a two-link ball pitching robot with the aim of throwing a ball as far as possible. The pitching robot is connected to a motor via a non-linear torsional spring at the shoulder joint. The elbow joint is passive and loaded with a linear torsional spring. We model the system based on an Euler–Lagrange formulation. Constraints on the motor torque and power as well as the angular velocity of the motor shaft are included in the model. By using an interior point method with gradients supplied by a discrete adjoint method, we numerically solve the resulting constrained control problem of finding the optimal piecewise constant motor torque profile and release position. Numerical experiments illustrate the effectiveness of our strategy as well as the effect of the constraints on the objective. In our experiments, the optimal motor torque gives rise to motions comprising an initial backswing; a transition, where the elbow spring accumulates potential energy; and finally a fast acceleration phase leading up the ball release.



## 1. Introduction

In sports medicine, biomechanics, and robotics, many researchers have studied motions utilizing dexterous actions. Several human sports such as baseball, handball, and javelin throw include nimble pitching. The overarm throws, in particular, involve one of the fastest joint rotations in the human body [1–3]. For distance throwing, the execution phase usually takes less than 1 s [4]. The dynamics of the human upper limb (including the shoulder, arm, and forearm) are in general non-linear and complicated to analyze. This is due to the large number of degrees of freedom associated with the upper limb and the three dimensional nature of its motions. However, for fast pitching motions, such as baseball throwing, the dynamics of a two-link robotic manipulator capture important aspects of the dynamics of human pitching [5–7]. In this study, we aim to find the optimal control of a two-link pitching robot. Thus we model a pitching robot that is constructed to capture some of the dynamics of the fast motions in human overarm pitching.

In the literature, there are many studies on motion control for optimal pitching with different objectives. For instance, Arisumi et al. [8] and Fagiolini et al. [9] proposed a strategy to control the swing motion of the manipulator. Kato et al. [10] studied throwing motion control to pitch an object towards a predefined target point for a two-degree of freedom robot. High-speed throwing motion for a two-link "hand–arm" system with three degrees of freedom is presented by Senoo et al. [11]. They introduced a kinetic chain approach to efficiently transmit power from body trunk to the throwing arm, forearm, and finally to the ball. To achieve an energy efficient throwing motion towards a desired direction, Katsumata et al. [5] proposed a deceleration control strategy based on output zeroing. They considered two types of two-link manipulators (the first model has a spring at the elbow joint and the other has a physical constraint at the elbow joint) to implement throwing motion control generated by rapid actions.

Mettin et al. [6] studied a two-link robot with an actuator at the shoulder joint and a passive spring loaded elbow joint. To maximize the horizontal velocity of the ball from a predefined position in the horizontal plane, they used a strategy to parameterize the robot motion in terms of geometric relations among the joint angles. Mettin and Shiriaev [12] reduced the complexity of the

* Corresponding author. Tel.: +46 90 786 6838; fax: +46 90 786 6126.
*E-mail addresses:* yedegl@cs.umu.se (E.L. Yedeg), eddiew@cs.umu.se (E. Wadbro).

0094-114X/$ – see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.mechmachtheory.2013.06.009

finite-time optimal control problem for an underactuated two-link robot and presented a control strategy based on a geometric parameterization. Kim et al. [4,13] studied whole-body pitching and optimized throwing motions of biped humanoid robots. Among many other related studies, Xu et al. [14] presented a golf swing controller for a two-link golf swing robot, and Senoo et al. [15] proposed a robotic batting algorithm. In the works mentioned above, the motion is typically parameterized using a modest number of control variables, such as coefficients in a polynomial, or amplitudes and frequencies. Such a parameterization simplifies the resulting optimization problem but also reduces the set of possible motions. Thus, even though this approach is useful in many cases, it may miss unforeseen superior controls.

In this work, we propose a direct approach to obtain the optimal control of the ball pitching robot. To the best of our knowledge, this problem has not been previously attacked directly using mathematical programming, as opposed to methods based on solving the optimality conditions. Our approach is easily generalizable to work with other underactuated mechanical systems, including more degrees of freedom describing the system dynamics and additional constraints. Moreover, it can be extended to include other objectives, such as pitching toward a pre-defined target point, as well as alternative control parameterizations.

Here, we model a two-link robot with the objective to throw a ball as far as possible from the shoulder joint position. We include constraints on the motor torque, power, and the angular velocity of the motor shaft into the problem formulation. For computational efficiency, we replace the time-global constraints on maximum allowed power and maximum angular velocity of the motor shaft by approximations based on integral quantities. The approximate constraints are expected to be slightly restrictive, and the original time-global constraints have been satisfied for all our numerical experiments. We numerically solve the resulting optimal control problem by using an interior point method with BFGS Hessian approximation and a discrete adjoint based gradient computation.

## 2. Problem statement

We study a two-link pitching robot with an active gripping mechanism, illustrated in Fig. 1. To simplify the discussion, we will henceforth denote the two links by the arm and forearm, respectively. The base of the robot, or the robot's shoulder, is held fixed at the origin of our coordinate system. At the end of the forearm there is a gripping mechanism able to hold a ball and release it at any specified time. The two links are connected at the elbow joint by a linear torsional spring. We let $q_2$ denote the angle change between the arm and forearm at the elbow joint. The driving force of the pitching motion is a motor; the motor shaft is connected to the arm at the shoulder joint by a non-linear torsional spring. The configuration of the arm and the motor shaft is described by the angles $q_1$ and $q_m$, measured with respect to the horizontal axis, respectively. To sum up, the pitching robot is mechanically a three-degree kinematic system plus a gripping mechanism. The main objective of this study is to find the time evolution of the motor torque $\tau(t)$ and the release time $t_r$ to throw a ball as far as possible.

### 2.1. Modeling the dynamics of the robotic system

Referring to Fig. 1, let $q = [q_m, q_1, q_2]^T$ be a vector of generalized coordinates of the robotic manipulator and let $\theta = [q, \dot{q}]^T \in \mathbb{R}^6$ be the state vector. By using the Euler–Lagrange formulation derived in A, we write the equations of motion of the pitching robot in state space representation as

$$\dot{\theta} = f(\theta, \tau). \tag{1}$$

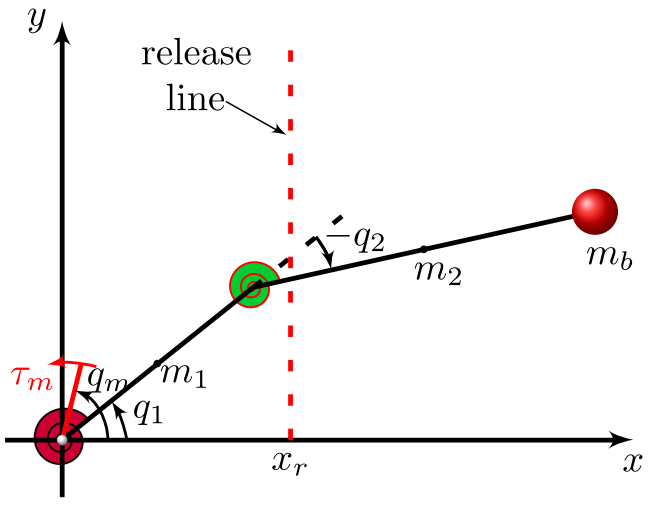


**Fig. 1.** Initial configuration of the pitching robot and the release line.

Here

$$f(\theta,\tau)=\begin{bmatrix}\theta_4\\ \theta_5\\ \theta_6\\ I_m^{-1}(\tau-\tau_s(\theta))\\ M(\theta)^{-1}\left(K(\theta)-G(\theta)-C(\theta)\begin{bmatrix}\theta_4\\ \theta_5\end{bmatrix}\right)\end{bmatrix}, \tag{2}$$

where $I_m$ is the motor inertia, $\tau_s(\theta)$ the torque due to the non-linear spring at the shoulder joint, $M(\theta)$ the inertia matrix, $C(\theta)$ the matrix of centrifugal and Coriolis forces, and the vectors $G(\theta)$ and $K(\theta)$ represent gravitational and spring forces, respectively. We emphasize that this rewrite of equation system (A.3) from Appendix A is possible since the inertia matrix $M(\theta)$ is invertible and $I_m > 0$. Here, we have only one control input, that is, we can control only the motor torque at the robot's shoulder. The two springs, connecting the arm with the forearm and motor shaft, respectively, are passive and we can only indirectly control them. Thus, the system above constitutes an underactuated mechanical system [16].

### 2.2. Pitching distance

To throw the ball as far as possible, we seek to determine the optimal motor torque $\tau$ as a function of time and to find the optimal position $x_r$ of the release line—the vertical line, $(x, y) = (x_r, s)$, $s \in \mathbb{R}$, at which the gripping mechanism of the robot releases the ball. The motor torque as a function of time, the position of the release line, and the initial configuration of the robot given at time $t_0 = 0$ determine the time evolution of the generalized coordinates, $q_m(t)$, $q_1(t)$, and $q_2(t)$, as well as the duration of the motion $t_r$. The trajectory of the ball during the pitching motion is determined by the time evolution of the generalized coordinates. Fig. 2 shows the trajectory of the ball during the pitching motion and the ball velocity $v_b$ at the release position.

As illustrated in Fig. 3, after the ball is released, it enters ballistic flight and finally hits the ground at the point $(J, 0)$. Note that we are pitching in the negative direction with respect to our coordinate system. So by minimizing $J$, we maximize the distance $D = |(J,0) - (0,0)|$ between the point where the ball hits the ground and the origin (the robot's shoulder).

We compute the distance thrown from the configuration of the pitching robot at the time of ball release by assuming there is no external force except from gravity. So, in particular, we neglect the air resistance. During ballistic flight, the ball moves at constant velocity in the horizontal direction,

$$\dot{x}_b(t_r) = -l_1\sin(q_1(t_r))\dot{q}_1(t_r) - l_2\sin(q_1(t_r)+q_2(t_r))(\dot{q}_1(t_r)+\dot{q}_2(t_r)), \tag{3}$$

where $l_1$ and $l_2$ are the lengths of the arm and forearm of the robot, respectively. In the vertical direction the ball accelerates toward the ground due to the gravitational force, thus the time evolution follows a parabola. The flight time of the ball, that is the time from release until it hits the ground, is

$$\frac{1}{g}\left(\dot{y}_b(t_r)+\sqrt{\dot{y}_b(t_r)^2+2gy_b(t_r)}\right), \tag{4}$$

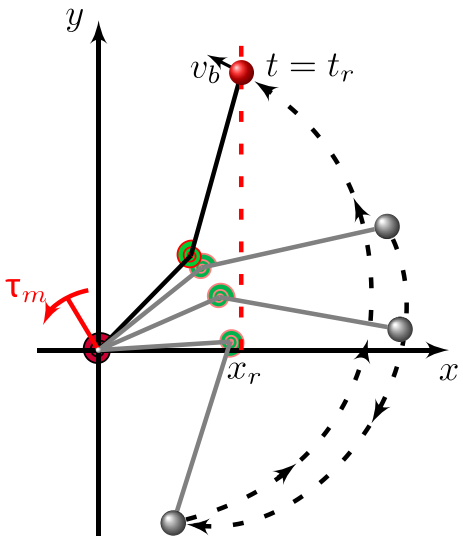


**Fig. 2.** Configuration of the pitching robot at the release time and the ball trajectory during the pitching motion.

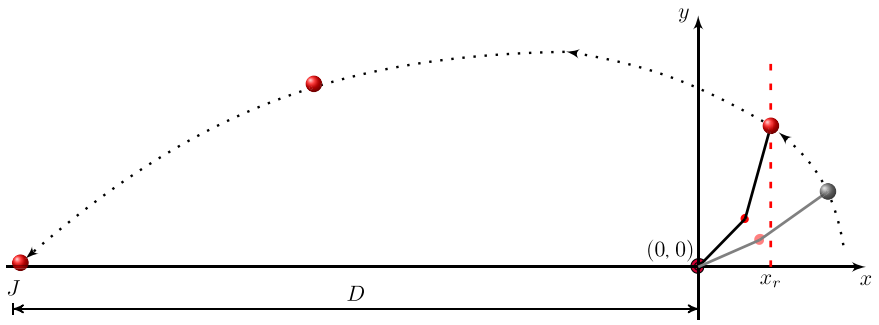


**Fig. 3.** Ball trajectory and the horizontal distance between the origin and the point where the ball hits the ground.

where $g$ is the gravitational constant,

$$y_b(t_r) = l_1 \sin(q_1(t_r)) + l_2 \sin(q_1(t_r) + q_2(t_r)) \tag{5}$$

is the vertical position of the ball at release time, and

$$\dot{y}_b(t_r) = l_1 \cos(q_1(t_r))\dot{q}_1(t_r) + l_2 \cos(q_1(t_r) + q_2(t_r))(\dot{q}_1(t_r) + \dot{q}_2(t_r)) \tag{6}$$

is the vertical velocity of the ball at release time. The total horizontal flight distance is found by multiplying the constant horizontal velocity with the flight time. Since the ball is released when it crosses the release line, we know that $x_b(t_r) = x_r$, where $x_b(t) = l_1 \cos(q_1(t)) + l_2 \cos(q_1(t) + q_2(t))$ is the horizontal position of the ball at time $t$. Hence, we conclude that the ball lands at the point $(J, 0)$, where

$$J = x_r + \frac{\dot{x}_b(t_r)}{g}\left(\dot{y}_b(t_r) + \sqrt{\dot{y}_b(t_r)^2 + 2g y_b(t_r)}\right). \tag{7}$$

### 2.3. Constraints

The system has four types of limits on the robot performance for all time $t \in [t_0, t_r]$. The absolute torque produced by the actuator has limited value $\overline{\tau}$, thus

$$|\tau| \le \overline{\tau}. \tag{8}$$

In addition, the torque change in time is limited by $C_\tau$, that is,

$$|\dot{\tau}| \le C_\tau, \tag{9}$$

and the angular velocity of the motor shaft is limited, giving

$$|\dot{q}_m| \le Q_{\max}, \tag{10}$$

where $Q_{\max}$ is the maximum angular velocity of the motor shaft. The maximum input power of the motor is also limited, hence

$$|\tau \dot{q}_m| \le P_{\max}, \tag{11}$$

where $P_{\max}$ is the maximum power of the motor due to its angular velocity.

In this study, we optimize over the torque change in time and thus we include constraints (8) and (9) directly in the definition of admissible controls. A straightforward discretization of the remaining two time-global constrains (10) and (11) on the angular velocity of the motor shaft and input power leads to a discrete problem with two (discrete) constraints per time step until the ball reaches the release line. It is computationally much easier and efficient to handle a few integral constraints than bounds everywhere on the states. Hence, we derive integral approximations of the time-global bound constrains on the state.

Constraints (10) and (11) can be written in the form

$$\|u\|_{L^\infty(0,t_r)} = \sup_{t\in(0,t_r)} |u(t)| \le u_{\max}. \tag{12}$$

To approximate the $L^\infty$ norm constraint above, we use a time-scaled version of the standard $L^p$-norm, namely

$$\|u\|_{\hat{L}^{p(0,t_r)}} = \left(\frac{1}{t_r}\int_0^{t_r} |u|^p \, \mathrm{d}t\right)^{1/p}, \tag{13}$$

where the inclusion of the time-scaling removes the dependence on the unknown end time $t_r$. By definition, the $\hat{L}_1$ norm of $u$ is the average of $|u|$, and $\|u\|_{\hat{L}^p}$ tends to $\|u\|_{L^\infty}$ as $p \to \infty$. However, for any function $u$, it holds that $\|u\|_{\hat{L}^{p(0,t_r)}} \le \|u\|_{L^\infty(0,t_r)}$. For our problem, we want to be able to bound the $L^\infty$ norm of $u$ from above by imposing a constraint on the $\hat{L}^p$ norm of $u$. If the absolute value of the function values of $u$ are uniformly distributed between 0 and $\|u\|_{L^\infty}$, we expect that

$$\begin{aligned}\|u\|_{\hat{L}^{p(0,t_r)}} &= \left(\frac{1}{t_r}\int_0^{t_r} |u|^p \, \mathrm{d}t\right)^{1/p} \\ &\approx \left(\frac{1}{t_r}\int_0^{t_r} \left|\|u\|_{L^\infty(0,t_r)}\frac{t}{t_r}\right|^p \, \mathrm{d}t\right)^{1/p} = \left(\frac{1}{p+1}\right)^{1/p}\|u\|_{L^\infty(0,t_r)}.\end{aligned} \tag{14}$$

When the constraint is active for this particular problem, we do not expect the function values of $|u|$ to be perfectly uniformly distributed between 0 and $\|u\|_{L^\infty}$ but denser close to $\|u\|_{L^\infty}$. In other words, we expect that the distribution of $|u|$ is negatively skew, which means that

$$\|u\|_{\hat{L}^{p(0,t_r)}} \gtrsim \left(\frac{1}{p+1}\right)^{1/p}\|u\|_{L^\infty(0,t_r)}. \tag{15}$$

Thus, in our numerical treatment of the problem, we approximate constraints (10) and (11) by

$$\|\theta^4\|_{\hat{L}^{p(0,t_r)}} \le \left(\frac{1}{p+1}\right)^{1/p} Q_{\max}, \tag{16a}$$

$$\|\tau\theta^4\|_{\hat{L}^{p(0,t_r)}} \le \left(\frac{1}{p+1}\right)^{1/p} P_{\max}. \tag{16b}$$

Under the above assumptions, a function satisfying constraints (16a) and (16b) will also satisfy the original constraints (10) and (11). We also note that as the negative skewness of the distribution grows, the $\hat{L}^p$-norm constraints become increasingly restrictive. However, important to keep in mind that the above arguments are heuristics, so we need to check whether the $L^\infty$ constraints are satisfied after optimization.

### 2.4. Control parameterization and optimization problem

As we mentioned in the previous subsection, we optimize over the time change of the torque. Here, we let $\eta$ be our control function and define the motor torque indirectly by

$$\begin{aligned}\dot{\tau} &= \eta, \\ \tau(0) &= \tau_0.\end{aligned} \tag{17}$$

The set of feasible controls is defined as

$$\mathcal{A} = \left\{\eta \in L^\infty \,\middle|\, \sup_t |\eta(t)| \le C_\tau,\ \sup_t \left|\tau_0 + \int_0^t \eta(s)\, \mathrm{d}s\right| \le \overline{\tau}\right\}, \tag{18}$$

ensuring that constraints (8) and (9) are satisfied for all $\eta \in \mathcal{A}$. By putting together the system dynamics discussed in Section 2.1, our objective from Section 2.2, and the constraints from Section 2.3, we arrive at the optimal control problem

$$\begin{aligned} \min_{\eta\in\mathcal{A},x_r} \quad & J(x_r,\theta(t_r)) \\ \text{subject to} \quad & \dot{\tau}=\eta,\ \ \tau(0)=\tau_{(0)}, \\ & \dot{\theta}=f(\theta,\tau),\ \ \theta(0)=\theta^{(0)}, \\ & \|\theta_4\|_{\hat{L}^p(0,t_r)} \le \left(\frac{1}{p+1}\right)^{1/p} Q_{\max}, \\ & \|\tau\theta_4\|_{\hat{L}^p(0,t_r)} \le \left(\frac{1}{p+1}\right)^{1/p} P_{\max} \end{aligned} \tag{19}$$

where $\theta^{(0)}$ holds the initial state of the system, $t_r$ is a release time, that is the time when the ball crosses the release line in the throwing direction, and $x_r = x_b(t_r)$ is horizontal position of the release line.

There are two general approaches to tackle optimal control problems related to trajectory optimization, namely, direct and indirect methods [17]. An indirect method transforms the problem into another form before solving it. Typically, Pontryagin's Maximum Principle [18] is used to find the necessary conditions for the existence of an optimum. This allows the original optimal control problem to be transformed into a boundary value problem, which can be solved using standard techniques for differential equations. On the other hand, the direct methods perform a search for the control function that optimizes the objective functional. It transforms the continuous problem (infinite dimensional) to a discrete (finite dimensional) nonlinear programming problem using some parameterization of the state and control variables. So before optimizing, the state and control variables are approximated using an appropriate function approximation like the piecewise constant parameterization. We use one of the direct methods to solve the optimal control problem (19) and its discretization is presented in the following section.

## 3. The discrete problem

### 3.1. Discretization

We let $T$ be a sufficiently large time such that $0 < t_r \le T$ for any reasonable motor torque profile and solve state Eq. (1) numerically on a uniform discretization of the fixed time interval $I = [0,T]$. Let $t_0 = 0$, and define

$$t_k = t_0 + k\Delta t, \quad k = 1,\ldots,N, \tag{20}$$

where the constant step size $\Delta t = T/N$ is chosen such that $t_N = T$. We approximate the torque $\tau$ by a piecewise constant function $\tau_d$ defined as

$$\tau_d(t) = \tau_k \ \text{ for } \ t_{k-1} \le t < t_k, \quad k = 1,2,\ldots,N. \tag{21}$$

We let $\Theta_k$ denote the approximation of the state variable $\theta(t)$ at time $t = t_k$ and denote the approximation of $\theta_i(t_k)$ by $\Theta_{k,i}$. To discretize state Eq. (1), we use Heun's method and the discrete state equation is

$$\Theta_0 = \theta(t_0), \tag{22a}$$

$$\tilde{\Theta}_k = \Theta_{k-1} + \Delta t f(\Theta_{k-1},\tau_k), k = 1,2,\ldots,N, \tag{22b}$$

$$\Theta_k = \Theta_{k-1} + \frac{\Delta t}{2}\left(f(\Theta_{k-1},\tau_k) + f\left(\tilde{\Theta}_k,\tau_k\right)\right), \quad k = 1,2,\ldots,N. \tag{22c}$$

The discrete state (Eqs. (22a), (22b) and (22c)) can be written in matrix form as

$$0 = \mathbf{F}(\boldsymbol{\Theta},\tau) = \begin{pmatrix} \mathbf{F}_1(\Theta_0,\Theta_1,\tau_1) \\ \vdots \\ \mathbf{F}_N(\Theta_{N-1},\Theta_N,\tau_N) \end{pmatrix}, \tag{23}$$

where $\boldsymbol{\Theta} = (\Theta_1, \ldots, \Theta_N)^T$, $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_N)^T$, and

$$\mathbf{F}_k(\Theta_{k-1},\Theta_k,\tau_k) = \Theta_{k-1} + \frac{\Delta t}{2}\left(f(\Theta_{k-1},\tau_k) + f\left(\tilde{\Theta}_k,\tau_k\right)\right) - \Theta_k, \tag{24}$$

for $k = 1, 2, \ldots, N$ and with $\tilde{\Theta}_k$ defined as in expression (22b).

We define our discrete control $\boldsymbol{\eta} = (\eta_1,\eta_2, \ldots,\eta_N)^T$ to be the vector of torque changes at times $t_{k-1}$, that is, $\eta_k = \tau_k - \tau_{k-1}$. Thus

$$\boldsymbol{\tau} = \tau_0 + A\boldsymbol{\eta}, \tag{25}$$

where $A$ is a lower triangular matrix with all diagonal and sub-diagonal entries equal to 1. We define the following discrete version of the set of feasible controls (18) as

$$\mathcal{A}_d = \left\{ \boldsymbol{\eta} \in l^\infty \mid \sup_k |\eta_k| \le \Delta t C_\tau, \quad \sup_k \left| \tau_0 + \sum_{i=1}^{k} \eta_i \right| \le \overline{\tau} \right\}. \tag{26}$$

To discretize the constraints on angular velocity and power, we use the trapezoidal rule to numerically evaluate the integrals in the $\hat{L}^p$ norm (13). The release time $t_r$ is generally not equal to any of our time discretization points $t_k$. Here, we choose $j$ such that $t_j$ is the last discrete time point before the gripping mechanism of the pitching robot crosses the release line in the throwing direction. That is,

$$j = \min_k \{ k \mid x_r - x_k \le 0, x_r - x_{k+1} > 0, y_{k+1} - y_k \ge 0 \}, \tag{27}$$

where $x_k$ and $y_k$ are the horizontal and vertical position of the ball at time $t_k$, respectively. We approximate the time at which the ball crosses the release line by using linear interpolation based on the ball positions at times $t_j$ and $t_{j+1}$. We thus define

$$\beta = \frac{x_r - x_j}{x_{j+1} - x_j}, \tag{28}$$

and approximate the time at which the ball crosses the release line by $t_r = t_j + \beta \Delta t$. We define a discrete version of the $\hat{L}^p$ norm (13) based on the trapezoidal rule as

$$\|\mathbf{u}\|_{\hat{L}_d^p}(0, t_r) = \left( \frac{\Delta t}{2 t_r} \sum_{k=0}^{j-1} \left( |u_k|^p + |u_{k+1}|^p \right) + \frac{\beta \Delta t}{2 t_r} \left( |u_j|^p + |u_r|^p \right) \right)^{1/p}, \tag{29}$$

where $\mathbf{u}$ is a vector with elements $u_k$, approximating $u(t_k)$ for $k = 0, \ldots, N$, and $u_r = \beta u_j + (1 - \beta) u_{j+1}$ is the approximation of $u(t_r)$ obtained by using linear interpolation based on the approximations $u_j$ and $u_{j+1}$ of $u$ at times $t_j$ and $t_{j+1}$. The discrete form of constraints (16a) and (16b) then reads

$$\|\boldsymbol{\Theta}_{\cdot,4}\|_{\hat{L}_d^p}(0, t_r) \le \left( \frac{1}{p+1} \right)^{1/p} Q_{\max}, \tag{30a}$$

$$\|\mathbf{P}\|_{\hat{L}_d^p}(0, t_r) \le \left( \frac{1}{p+1} \right)^{1/p} P_{\max}, \tag{30b}$$

where $\boldsymbol{\Theta}_{\cdot,4}$ is the vector with elements $\Theta_{k,4}$, $k = 1, 2, \ldots, N$ and $\mathbf{P}$ is the vector with elements $\tau_k \Theta_{k,4}$, $k = 1, 2, \ldots, N$. By introducing the function

$$G(\boldsymbol{\Theta}, \boldsymbol{\tau}) = \begin{bmatrix} \|\boldsymbol{\Theta}^4\|_{\hat{L}_d^p}(0, t_r) - \left( \frac{1}{p+1} \right)^{1/p} Q_{\max} \\ \|\mathbf{P}\|_{\hat{L}_d^p}(0, t_r) - \left( \frac{1}{p+1} \right)^{1/p} P_{\max} \end{bmatrix}, \tag{31}$$

constraints (30a) and (30b) can be written as $G(\boldsymbol{\Theta}, \boldsymbol{\tau}) \le 0$.

The discrete objective function is given by

$$\tilde{J}(x_r, \boldsymbol{\Theta}) = J(x_r, \theta_r), \tag{32}$$

where $\boldsymbol{\Theta}$ solves the state Eq. (23) and $\theta_r = \beta \Theta_j + (1 - \beta) \Theta_{j+1}$ in which $j$ and $\beta$ are defined in expressions (27) and (28), respectively. Hence, a discrete version of the continuous optimal control problem (19) is

$$\begin{aligned} \min_{\boldsymbol{\eta} \in \mathcal{A}_d, x_r} \quad & \tilde{J}(x_r, \boldsymbol{\Theta}) \\ \text{subject to} \quad & \mathbf{F}(\boldsymbol{\Theta}, \tau) = 0, \\ & \tau = \tau_0 + A\boldsymbol{\eta}, \\ & \Theta_0 = \theta^{(0)}, \\ & G(\boldsymbol{\Theta}, \tau) \le 0. \end{aligned} \tag{33}$$

We solve the nonlinear optimal control problem (33) numerically by using an interior point method with BFGS Hessian approximation. Therefore, it is necessary to compute the gradients of the objective function and the constraint functions. There are different ways to estimate the gradients in the discrete case, for an overview we refer the article by Griesse and Walther [19]. In this

study, we use adjoint based gradient computation. In adjoint based gradient computation, the gradient of the objective function with respect to the control variable is expressed in terms of the state and adjoint equations.

### 3.2. Gradient computation based on adjoint method

In this section, we derive an expression for the gradient of $\tilde{J}$ with respect to $\boldsymbol{\eta}$. From discrete control problem (33), we know that $\Theta$ depends on $\tau$ which in turn depends on the control $\eta$. We perform gradient computation in two steps, first, we compute $\nabla_\tau \tilde{J}$, the gradient of $\tilde{J}$ with respect to $\boldsymbol{\tau}$, as described below. Then, we compute $\nabla_{\boldsymbol{\eta}} \tilde{J}$, the gradient of $\tilde{J}$ with respect to $\boldsymbol{\eta}$, by using $\nabla_\tau \tilde{J}$, expression (25), and the chain rule.

By the chain rule, the gradient of the objective function with respect to $\boldsymbol{\tau}$ can be written as

$$\nabla_{\boldsymbol{\tau}} \tilde{J} = \frac{\partial \tilde{J}}{\partial \boldsymbol{\Theta}} \frac{d\boldsymbol{\Theta}}{d\boldsymbol{\tau}}. \tag{34}$$

Differentiating the state Eq. (23) by using chain rule yields

$$\left(\frac{\partial \mathbf{F}}{\partial \boldsymbol{\Theta}}\right) \frac{d\boldsymbol{\Theta}}{d\boldsymbol{\tau}} + \frac{\partial \mathbf{F}}{\partial \boldsymbol{\tau}} = 0. \tag{35}$$

Let $\boldsymbol{\lambda} = (\lambda_0, \ldots, \lambda_N)^T$, with $\lambda_k \in \mathbb{R}^6$, be the solution of the adjoint equation

$$\boldsymbol{\lambda}^T \left(\frac{\partial \mathbf{F}}{\partial \boldsymbol{\Theta}}\right) = \frac{\partial \tilde{J}}{\partial \boldsymbol{\Theta}}. \tag{36}$$

The derivative matrix $\frac{\partial \mathbf{F}}{\partial \boldsymbol{\Theta}}$ is a lower bidiagonal matrix, thus we solve the adjoint Eq. (36) backward in time as

$$\begin{aligned} \lambda_N &= \frac{\partial \tilde{J}}{\partial \Theta_N} \\ \lambda_k &= \frac{\partial \tilde{J}}{\partial \Theta_k} - \lambda_{k+1} \frac{\partial \mathbf{F}_{k+1}}{\partial \Theta_k}, \quad k = N-1, \ldots, 0. \end{aligned} \tag{37}$$

By multiplying Eq. (35) with $\boldsymbol{\lambda}^T$, using the $\boldsymbol{\lambda}$ that solves the adjoint Eq. (36), we obtain that

$$\frac{\partial \tilde{J}}{\partial \boldsymbol{\Theta}} \frac{d\boldsymbol{\Theta}}{d\boldsymbol{\tau}} + \boldsymbol{\lambda}^T \frac{\partial \mathbf{F}}{\partial \boldsymbol{\tau}} = 0. \tag{38}$$

The first term of Eq. (38) equals the gradient of $\tilde{J}$ with respect to $\boldsymbol{\tau}$, and thus we conclude that

$$\nabla_{\boldsymbol{\tau}} \tilde{J} = -\boldsymbol{\lambda}^T \frac{\partial \mathbf{F}}{\partial \boldsymbol{\tau}}. \tag{39}$$

By using similar arguments for the components of the state constraint $G = (G_1, G_2)^T$ defined in expression (31), we obtain for $i = 1, 2$ that

$$\nabla_{\boldsymbol{\tau}} G_i = \frac{\partial G_i}{\partial \boldsymbol{\tau}} - \boldsymbol{\gamma_i} \frac{\partial \mathbf{F}}{\partial \boldsymbol{\tau}}, \tag{40}$$

where $\boldsymbol{\gamma_i}$ is the solution of

$$\boldsymbol{\gamma_i}^T \left(\frac{\partial \mathbf{F}}{\partial \boldsymbol{\Theta}}\right) = \frac{\partial G_i}{\partial \boldsymbol{\Theta}}. \tag{41}$$

## 4. Numerical experiment

In this section, we present the numerical results of the discrete optimal control problem discussed at the previous section. Here, we use parameters for a robot placed at Deutsche Zentrum für Luft- und Raumfahrt (DLR) in Munich. Table 1 presents these parameters as well as the mechanical constraints on power and velocity of the motor shaft and the input torque for this particular throwing robot. The non-linear spring at the shoulder produces a torque, which is approximated by the following function

$$\tau_s = a_3 (q_m - q_1)^3 + a_2 (q_m - q_1)^2 + a_1 (q_m - q_1) + a_0, \tag{42}$$

**Table 1**
Physical parameters and constraints of a ball throwing robot placed at Deutsche Zentrum für Luft- und Raumfahrt (DLR) in Munich.

| Parameter | Arm | Forearm |
|---|---|---|
| Length [m] | $l_1 = 0.3$ | $l_2 = 0.542$ |
| Distance to center of mass (COM) [m] | $l_{1c} = 0.2071$ | $l_{2c} = 0.2717$ |
| Mass [kg] | $m_1 = 2.934$ | $m_2 = 1.1022$ |
| Moment of inertia about COM [kgm$^2$] | $I_1 = 0.2067$ | $I_2 = 0.1362$ |
| Mass of the ball [kg] | $m_b = 0.064$ | |
| Gravitational constant [m/s$^2$] | $g = 9.81$ | |
| Motor inertia [kgm$^2$] | $I_m = 0.933739$ | |
| Spring constant at elbow joint [Nm/rad] | $k_2 = 14.1543$ | |
| Torque constraint [Nm] | $\lvert\tau\rvert \le 180$ | |
| Velocity constraint [rad/s] | $\lvert\dot{q}_m\rvert \le 3.787$ | |
| Power constraint [Nm/s] | $\lvert\tau\dot{q}_m\rvert \le 270$ | |

that is, a cubic polynomial in the angle between the motor shaft and the robot arm. The coefficients

$$a_0 = -0.0334, \quad a_1 = 77.8080, \quad a_2 = 2.9755, \quad \text{and } a_3 = 7761.7, \tag{43}$$

are obtained from a least-squares fit with measured data for the pitching robot we study in the range $|q_m - q_1| \le 0.25$ rad. In this range, the relative error between the measured data and the cubic model is less than $2.5 \times 10^{-3}$.

The initial conditions are determined such that the system is stationary at time $t_0 = 0$. Thus, given $q_1(0) = \pi - 2.234$ rad, we choose the initial conditions of the robot as

$$q_0 = [0.908, -0.172, 0.978] \text{ rad and } \dot{q}_0 = 0 \text{ rad/s}, \tag{44}$$

and set the initial motor torque $\tau_0$ to be 8.213 Nm.

We solve the discrete optimal control problem (33) numerically by using Matlab's function `fmincon` with the interior point option and a BFGS Hessian approximation. For numerical reasons, we impose the following natural constraint on the release line position

$$|x_r| \le l_1 + l_2, \tag{45}$$

where $l_1$ and $l_2$ are the length of the robot's arm and forearm, respectively.

For the numerical experiments presented in this section, we set $T = 2$ s, $\Delta t = 0.0025$ s, and $C_\tau = 1000$ Nm/s. Choosing $p$ in the $\hat{L}^p$ norm approximating the $L^\infty$ norm for the constraints on angular velocity and power is a balance act. That is, we want to have $p$ as large as possible to obtain a better approximation of the original $L^\infty$-norm constraints but not too large to cause problems in the numerical computations. In our early experiments, we used a relatively small value for parameter $p$, for instance 4 and 6. But, we observed that for these values of $p$, the approximative $\hat{L}^p$ norm constraints were overly restrictive. That is, the ratio between the $\hat{L}^p$ norm and the $L^\infty$ norm of the constraint functions was significantly larger than $1/(p+1)^{1/p}$. By performing additional experiments and analysis, taking into account that the computations are performed in double precision (IEEE binary64), we found that $p = 100$ offers a good tradeoff for this particular problem. Thus, all experiments presented in this section use $p = 100$.

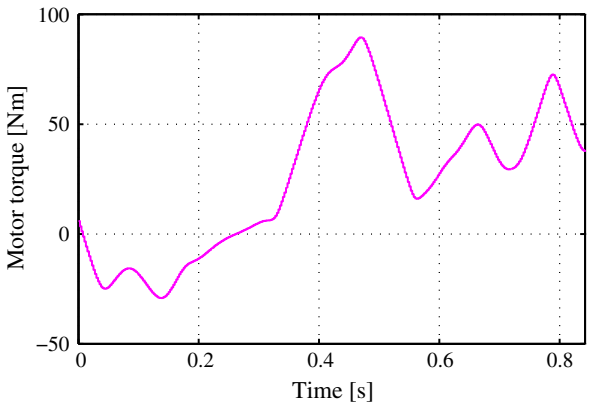


**Fig. 4.** Optimal torque obtained by solving problem (32) with parameters for the pitching robot presented in Table 1.

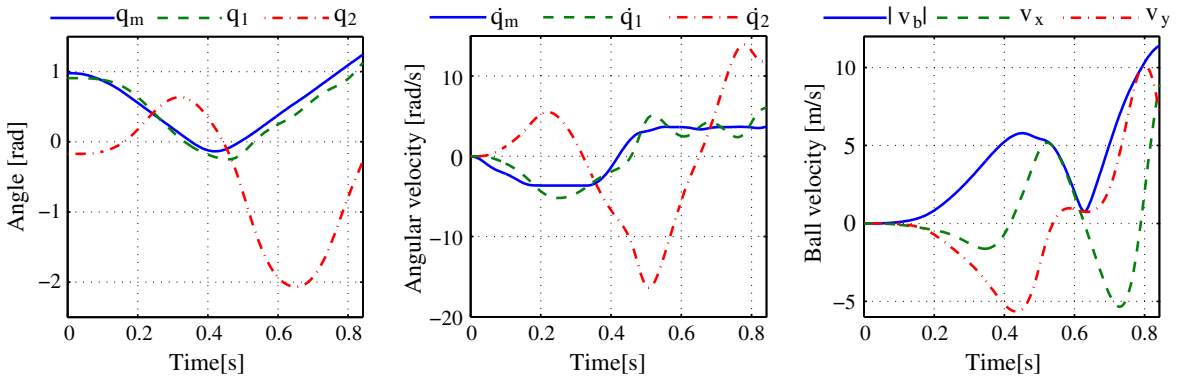


**Fig. 5.** Time evolution of the generalized coordinates $q_1$, $q_2$, and $q_m$ (left), the corresponding angular velocities $\dot{q}_1, \dot{q}_2$ and $\dot{q}_m$ (middle), and the velocity of the ball (right) computed using the optimal input torque from Fig. 1.

In general, it may be hard or even impossible to find a feasible starting point when dealing with multiple non-linear constraints. Fortunately, for the studied problem we can directly identify a feasible control—any sufficiently slow pitching motion is feasible. Here, we let the initial torque change profile be given as

$$\eta_k = \begin{cases} 1/6, & \text{for } k = 10, 11, \ldots, 80, \\ 0, & \text{otherwise.} \end{cases} \tag{46}$$

The reason for setting the torque change to zero for the first few time steps is that we do not want to put any a priori preference on the starting direction of the pitching motion. The initial release line is chosen to be $x_r = l_1 \cos(q_1(0)) + l_2 \cos(q_1(0) + q_2(0)) - 0.025$. With the aforementioned initial conditions all the constraints in the optimal control problem (33) and the constraints in Table 1 are satisfied.

Fig. 4 shows the time evaluation of the resulting optimal input torque profile $\tau_d$, computed from the resulting optimal control $\boldsymbol{\eta}$ using expression (25). The torque attains its peak value 89.54 Nm after 0.473 s. For the studied setup, the optimal release line is located at $x_r = 0.506$ m from the origin and the total duration of the pitching motion is $t_r = 0.843$ s. The ball is thrown at an upward angle of 40.82° with velocity of 11.36 m/s. After the ball is released, it follows a ballistic trajectory and finally hits the ground 13.25 m away from the origin (the robot's shoulder position).

Fig. 5 shows the time evolution, from time $t_0$ to $t_r$, of the generalized coordinates $q_1$, $q_2$, and $q_m$ (left diagram), the angular velocities $\dot{q}_1, \dot{q}_2$ and $\dot{q}_m$ (middle diagram), and the velocity of the ball (right diagram). Recall that $q_2$ is measured with respect to the arm extension line, while $q_1$ and $q_m$ are measured with respect to the horizontal line. The left diagram illustrates, in particular,

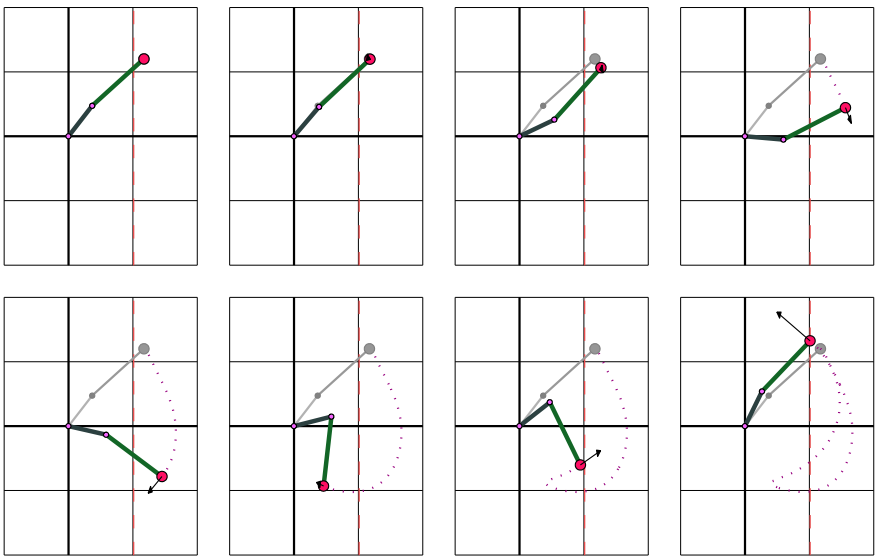

**Fig. 6.** Snapshots of the ball pitching robot during the throwing motion obtained by using the optimal input torque presented in Fig. 4. The snapshots are taken at times (top row from left to right) $t = 0, t_r/7, 2t_r/7$, and $3t_r/7$ and (bottom row from left to right) $t = 4t_r/7, 5t_r, 6t_r/7$, and $t_r$, where $t_r = 0.843$ s is the duration of the throwing motion.

**Table 2**
Results obtained by solving the optimal control problem (**32**) for different bounds on the angular velocity $Q_{\max}$, keeping the constraints on maximum power, $P_{\max} = 270$ Nm/s, and maximum absolute torque, $\overline{\tau} = 180$ Nm, unchanged.

| $Q_{\max}$ [rad] | $\max(\lvert\dot{q}_m\rvert)$ [rad/s] | $\max(\lvert\tau\rvert)$ [Nm] | $\max(\lvert\tau\dot{q}_m\rvert)$ [Nm/s] | $\lvert v_b(t_f)\rvert$ [m/s] | Release angle [°] | D [m] |
|---|---|---|---|---|---|---|
| 3.787 | 3.717 | 89.537 | 264.617 | 11.364 | 40.82 | 13.25 |
| 4 | 3.907 | 75.718 | 223.110 | 11.689 | 40.94 | 14.02 |
| 5 | 4.876 | 83.982 | 262.915 | 13.606 | 41.73 | 18.96 |
| 6 | 5.867 | 92.137 | 262.673 | 15.201 | 42.79 | 23.61 |
| 7 | 6.484 | 99.619 | 262.415 | 16.128 | 43.52 | 26.55 |
| 8 | 6.649 | 107.756 | 263.005 | 16.756 | 44.83 | 28.64 |
| 9 | 6.421 | 96.028 | 261.963 | 16.163 | 44.06 | 26.68 |

that $|q_m - q_1| \leq 0.25$ rad throughout the motion; hence, the cubic model for the non-linear spring connecting motor shaft and arm at the shoulder is valid. From the middle diagram of Fig. 5, it can be seen that the angular velocity $\dot{q}_m$ of the motor shaft (solid line) essentially has four phases. Starting from a stationary position, it first decelerates until it reaches an angular velocity of about $-3.7$ rad/s. In the second phase, the motor shaft moves at an almost constant angular velocity. This is followed by a fast acceleration comprising the third phase. Finally, the motor shaft moves at a relatively constant angular velocity of about 3.7 rad/s. During the last phase, the angular velocity $\dot{q}_2$ of the forearm with respect to the arm (dashed-dotted line) increases from $-16.39$ rad/s to its maximum 13.90 rad/s, and then it starts to decrease to 11.72 rad/s. When $\dot{q}_2$ starts to decrease, the angular velocity of the arm $\dot{q}_1$ (dashed line) starts to increase from 2.38 rad/s to its maximum 6.06 rad/s. In total, the angular velocity $\dot{q}_1 + \dot{q}_2$ of the forearm measured with respect to the horizontal line is strictly increasing during the last 0.35 s of the pitching motion. These rapid increments of angular velocities $\dot{q}_1$ and $\dot{q}_2$ accelerate the ball very fast as time closes to releasing, and they maximize the distance thrown. The right diagram in Fig. 5 shows the absolute velocity $|v_b|$ (solid line) of the ball as well as its horizontal component $v_x$ (dash-dotted line) and vertical component $v_y$ (dashed line). The maximum absolute velocity 11.36 m/s is attained at the release time.

Fig. 6 shows snapshots of the robot and illustrates the trajectory of the ball during the throwing motion. The snapshots are taken at equal time interval. The top row shows the configuration of the robot and the ball trajectory at times, from left to right, $t = 0$, $t_r/7$, $2t_r/7$, and $3t_r/7$ and the bottom row shows the robot at times, from left to right, $t = 4t_r/7$, $5t_r/7$, $6t_r/7$, and $t_r$, where $t_r = 0.843$ s is the duration of the throwing motion. The snapshots suggest that a significant part (about three quarters of the total pitching time) of the throwing time is spent on swinging the ball backward against the throwing direction to get longer interval of acceleration. After about two-thirds of the pitching time, the motor shaft and the robot's arm change direction and start to move in the throwing direction. While the forearm still moves against the throwing direction and the arm moves forward, the spring at the elbow joint accumulates potential energy. In the last quarter of the pitching time, the arm and forearm swings forward to the direction of the throw and the ball accelerates rapidly and reaches its maximum velocity at the release. The maximum absolute angular velocity of the motor shaft is 3.717 rad/s and the maximum power is 264.412 Nm/s, that is, the original $L^\infty$ constraints are satisfied. For the current setup, the torque constraint is not active; in fact the absolute torque is about half as large as the maximum allowed. By performing additional experiments, changing the constraints on the angular velocity of the motor shaft and the input power, we observed that, for the current setup, the angular velocity constraint on the motor shaft is the most restrictive one.

Table 2 and Fig. 7 present results obtained by solving control problem (33) for different bounds on the angular velocity, $Q_{max}$, and keeping all other constraints unchanged. In particular, Table 2 shows that the original time-global constraints are satisfied for all experiments. This table also shows the absolute release velocity, the release angle, and the distance thrown by the pitching robot. Fig. 7 illustrates the optimal release lines, the release velocities, and the trajectories of the ball obtained by using the

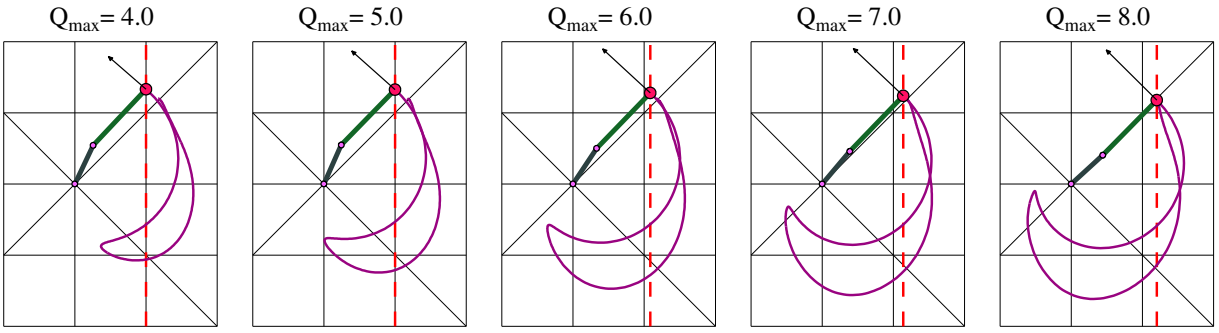


**Fig. 7.** Ball trajectories during pitching motion for different $Q_{\max}$, the bound on the angular velocity, keeping the constraints on maximum power, $P_{\max} = 270$ Nm/s, and maximum absolute torque, $\overline{\tau} = 180$ Nm, unchanged.

resulting optimal input torques. We observe that a small increment of $Q_{\max}$ has a significant effect on the distance thrown, at least when it is close to our reference $Q_{\max} = 3.787$ rad/s. Increasing the maximum admissible angular velocity of the motor shaft, allows the system to have a longer back-swing and acceleration period. This yields a significantly higher ball velocity at the release line and thus longer distances are thrown. More precisely, increasing $Q_{\max}$ from 4 rad/s to 5 rad/s and from 5 rad/s to 6 rad/s increases the distance thrown by 4.94 m and 4.65 m respectively. As $Q_{\max}$ grows, the power constraint gets more restrictive, for example, increasing $Q_{\max}$ from 6 rad/s to 7 rad/s and from 7 rad/s to 8 rad/s increases the distance thrown by 2.94 m and 2.09 m respectively. Moreover, for $Q_{\max} \gtrsim 7$ rad/s the constraint on maximum angular velocity of the motor shaft becomes inactive. This entails that there are multiple local minima for the problem without constraint on the maximum angular velocity of the motor shaft. In our experiments, we have noticed that even though the optimal control changes, the resulting motions share the same characteristics and the performance for all these controls is comparable.

## 5. Conclusions

This work investigates the problem of making a two-link pitching robot throw a ball as far as possible. The pitching robot we model is placed at DLR, Munich, and is constructed to capture the important aspects of upper limb dynamics of human overarm throws. A motor is connected to the robot with a non-linear torsional spring at the shoulder joint. By using an Euler–Lagrange formulation for the system dynamics and including constraints on the motor torque, motor power, and the angular velocity of the motor shaft, we arrive at a constrained optimal control problem with unknown end-time.

The constraints are required to hold at all times during the pitching motion. Here, we include the constraint on torque and torque change in our parameterization and approximate the $L^\infty$ constrains limiting the motor power and motor shaft velocity by using a time-scaled version of the standard $L^p$ norm together with a non-linear scaling of the actual constraint value. Under relatively non-restrictive assumptions, it can be shown that a function satisfying the approximate constrains will also satisfy the original $L^\infty$ constraints. In other words, the approximate constraints are likely to be overly restrictive (as is illustrated in Table 2 which shows that the original time-global constraints are satisfied for all experiments). In the discrete case, this approximation reduces the number of constraints with factor equal to the number of time steps taken in the discrete solution of the state equation.

After discretization, the optimization problem consists of finding the optimal release position and the optimal piecewise constant motor torque profile. By using parameters of the robot at DLR, we solve the control problem and noticed that for this particular setup, the constraint on motor shaft velocity was the most restrictive. The resulting optimal input torque generates a motion that starts with swinging the ball against the throwing motion allowing the spring at the elbow joint to accumulate potential energy. The back-swing takes about three quarters of the total time of the pitching motion, during the last quarter of the pitching time, the robot's arm and forearm accelerates hastily until the ball is released. For the studied setup, we observed that the angular velocity constraint on the motor shaft is the most restrictive.

The method presented in this paper is general and can easily be extended to include more complex dynamics and constraints. An additional interesting problem to study would be to use the current ball-pitching robot without the gripping mechanism at the end of the forearm. By replacing the gripping mechanism with a dish, essentially yielding a catapult, additional problems arise. For example, since the release happens spontaneously, we cannot stipulate an exact release line or time in the same manner as in the current contribution. Moreover, it is no longer possible to have a long back-swing since the ball would be dropped once the dish tilts too much in any direction.

## Acknowledgments

The authors are grateful to Uwe Mettin and Anton Shiriaev for introducing us to this interesting problem and providing valuable discussions. We thank Deutsche Zentrum für Luft- und Raumfahrt in Munich, for sharing measured data and parameters of their ball pitching robot. Finally, we are glad to thank Martin Berggren for fruitful discussions and for providing constructive feedback on the manuscript.

## Appendix A. Lagrangian and equation of motion

Consider the two-link throwing robot illustrated in Fig. 1 and let $K_m$, $K_1$, $K_2$, and $K_b$ denote the kinetic energies of the motor, arm, and forearm, respectively. Let $P_1$, $P_2$, and $P_b$ denote the potential energies of the arm, forearm, and ball, respectively. Further, denote the potential energies of the springs at the shoulder and elbow joints by $P_{S1}$, and $P_{S2}$, respectively. The Lagrangian of the system is thus

$$\mathcal{L} = (K_m + K_1 + K_2 + K_b) - (P_1 + P_2 + P_b + P_{S1} + P_{S2}). \tag{A.1}$$

The Euler–Lagrange formulation governing the dynamical system of the throwing robot is

$$\frac{d}{dt}\left[\frac{\partial \mathcal{L}}{\partial \dot{q}_m}\right] - \frac{\partial \mathcal{L}}{\partial q_m} = \tau, \quad \frac{d}{dt}\left[\frac{\partial \mathcal{L}}{\partial \dot{q}_1}\right] - \frac{\partial \mathcal{L}}{\partial q_1} = 0, \quad \frac{d}{dt}\left[\frac{\partial \mathcal{L}}{\partial \dot{q}_2}\right] - \frac{\partial \mathcal{L}}{\partial q_2} = 0. \tag{A.2}$$

The above system can be written as

$$\begin{aligned} &I_m\ddot{q}_m + \tau_s = \tau, \\ &M\begin{bmatrix}\ddot{q}_1\\ \ddot{q}_2\end{bmatrix} + C\begin{bmatrix}\dot{q}_1\\ \dot{q}_2\end{bmatrix} + G + K = 0, \end{aligned} \tag{A.3}$$

where $I_m$ is the motor inertia, $\tau_s$ is the torque due to the nonlinear spring at the shoulder joint, the symmetric positive definite inertia matrix

$$M = \begin{bmatrix} p_1 + p_2 + 2p_3\cos(q_2) & p_2 + p_3\cos(q_2) \\ p_2 + p_3\cos(q_2) & p_2 \end{bmatrix}, \tag{A.4}$$

the matrix of centrifugal and Coriolis forces

$$C = p_3\cos(q_2)\begin{bmatrix} -\dot{q}_2 & -\dot{q}_2 - \dot{q}_1 \\ \dot{q}_1 & 0 \end{bmatrix}, \tag{A.5}$$

the vector of gravitational forces

$$G = \begin{bmatrix} p_4\cos(q_1) + p_5\cos(q_1+q_2) \\ p_5\cos(q_1+q_2) \end{bmatrix}, \tag{A.6}$$

and the vector of spring forces

$$K(q) = \begin{bmatrix} -\tau_s \\ k_2 q_2 \end{bmatrix}, \tag{A.7}$$

in which $k_2$ is the spring constant for the linear spring at the elbow joint. Parameters $p_1$, $p_2$, $p_3$, $p_4$, and $p_5$ are constructed from the physical parameters of the robot (given in Table 1) as

$$\begin{aligned} &p_1 = I_1 + m_1 l_{c1}^2 + m_2 l_1^2 + m_b l_1^2, \quad p_2 = I_2 + m_2 l_{c2}^2 + m_b l_2^2, \\ &p_3 = m_2 l_1 l_{c2} + m_b l_1 l_2, \qquad\qquad p_4 = m_1 g l_{c1} + m_2 g l_2 + m_b g l_1, \quad \text{and} \\ &p_5 = m_2 g l_{c2} + m_b g l_2. \end{aligned} \tag{A.8}$$

## References


[1] S.L. Werner, G.S. Fleisig, C.J. Dillman, J.R. Andrews, Biomechanics of the elbow during baseball pitching, The Journal of Orthopaedic and Sports Physical Therapy 17 (1993) 274–278.
[2] V. Zatsiorsky, Biomechanics in Sport: Performance Enhancement and Injury Prevention: Olympic Encyclopaedia of Sports Medicine, 2000.
[3] R. Bartlett, Throwing: fundamentals and practical applications, 18 International Symposium on Biomechanics in Sports, 2000, pp. 1–8.
[4] J.H. Kim, Y. Xiang, J. Yang, J.S. Arora, K. Abdel-Malek, Dynamic motion planning of overarm throw for a biped human multibody system, Multibody System Dynamics 24 (2010) 1–24.
[5] S. Katsumata, S. Ichinose, T. Shoji, S. Nakaura, M. Sampei, Throwing motion control based on output zeroing utilizing 2-link underactuated arm, in: Proceedings of the 2009 conference on American Control Conference, 2009, pp. 3057–3064.
[6] U. Mettin, A.S. Shiriaev, L.B. Freidovich, M. Sampei, Optimal ball pitching with an underactuated model of a human arm, IEEE International Conference on Robotics and Automation, ICRA, 2010, pp. 5009–5014.
[7] S. Ichinose, S. Katsumata, S. Nakaura, M. Sampei, Throwing motion control experiment utilizing 2-link arm passive joint, SICE Annual Conference, 2008, pp. 3256–3261.
[8] H. Arisumi, T. Kotoku, K. Komoriya, A study of casting manipulation (swing motion control and planning throwing motion), IEEE International Conference on Intelligent Robots and Systems, vol. 1, 1997, pp. 168–174.
[9] A. Fagiolini, H. Arisumi, A. Bicchi, Visual-based feedback control of casting manipulation, in: Proceedings of the 2005 IEEE International Conference on Robotics and Automation, 2005, pp. 2191–2196.
[10] N. Kato, K. Matsuda, T. Nakamura, Adaptive control for a throwing motion of a 2 DOF robot, International Workshop on Advanced Motion Control, vol. 1, 1996, pp. 203–207.
[11] T. Senoo, A. Namiki, M. Ishikawa, High-speed throwing motion based on kinetic chain approach, International Conference on Intelligent Robots and Systems, 2008, pp. 3206–3211.
[12] U. Mettin, A.S. Shiriaev, Ball-pitching challenge with an underactuated two-link robot arm, 18th World Congress of the International Federation of Automatic Control, vol. 18, 2011, pp. 11399–11404.
[13] J.H. Kim, Optimization of throwing motion planning for whole-body humanoid mechanism: sidearm and maximum distance, Mechanism and Machine Theory 46 (2011) 438–453.
[14] C. Xu, A. Ming, T. Nagaoka, M. Shimojo, Motion control of a golf swing robot, Journal of Intelligent and Robotic Systems 56 (2009) 277–299.
[15] T. Senoo, A. Namiki, M. Ishikawa, High-speed batting using a multi-jointed manipulator, in: Proceedings of 2004 IEEE International Conference on Robotics and Automation, 2004, pp. 1191–1196.
[16] M.W. Spong, M.W. Spong, Underactuated mechanical systems, Control problems in robotics and automation, Lecture Notes in Control and Information Sciences, vol. 230, 1998, pp. 135–150.
[17] O. von Stryk, R. Bulirsch, Direct and indirect methods for trajectory optimization, Annals of Operations Research 37 (1992) 357–373.
[18] L.S. Pontryagin, V.G. Boltyanskii, R.V. Gamkrelidze, E.F. Mishchenko, The Mathematical Theory of Optimal Processes, Wiley, 1962.
[19] R. Griesse, A. Walther, Evaluating gradients in optimal control: continuous adjoints versus automatic differentiation, Journal of Optimization Theory and Applications 122 (2004) 63–86.