

Constrained Inverse Optimal Control With Application to a Human Manipulation Task

Journal Article**Author(s):**

Menner, Marcel; Worsnop, Peter; Zeilinger, Melanie N.

Publication date:

2021-03

Permanent link:

<https://doi.org/10.3929/ethz-b-000387818>

Rights / license:

In Copyright - Non-Commercial Use Permitted

Originally published in:

IEEE Transactions on Control Systems Technology 29(2), <https://doi.org/10.1109/tcst.2019.2955663>

Funding acknowledgement:

157601 - Safety and Performance for Human in the Loop Control (SNF)

Constrained Inverse Optimal Control with Application to a Human Manipulation Task

Marcel Menner, Peter Worsnop, and Melanie N. Zeilinger

Abstract—This paper presents an inverse optimal control methodology and its application to training a predictive model of human motor control from a manipulation task. It introduces a convex formulation for learning both objective function and constraints of an infinite-horizon constrained optimal control problem with nonlinear system dynamics. The inverse approach utilizes Bellman’s principle of optimality to formulate the infinite-horizon optimal control problem as a shortest path problem and Lagrange multipliers to identify constraints. We highlight the key benefit of using the shortest path formulation, i.e. the possibility of training the predictive model with short and selected trajectory segments. The method is applied to training a predictive model of movements of a human subject from a manipulation task. The study indicates that individual human movements can be predicted with low error using an infinite-horizon optimal control problem with constraints on shoulder movement.

Index Terms—Imitation learning, learning for dynamics and control, learning from demonstrations, manipulation tasks.

I. INTRODUCTION

As robotic systems are applied to increasingly unstructured and unpredictable environments, the ability to identify and adapt to their environment is becoming of critical importance. The collaboration with humans represents a particular challenge, as the interaction varies between individuals. The manipulation of an articulated object by a human in collaboration with a robot is one example, where the robot performance can be improved by learning a model to describe and predict the human motor control behavior [1].

The literature on human control behavior widely agrees on the fact that human motor performance is achieved through a reactive and a predictive component, cf. the review in [2]. The reactive component is triggered by sensory inputs and updates an ongoing motor command; it can therefore be interpreted as feedback control action. The predictive component capitalizes on the ability to anticipate motor events based on memory in order to accomplish a given task under foreseeable conditions, which can be interpreted as feedforward action [3]. The existence of these two components has been highlighted in studies of various motor control tasks, including grasping and manipulation [4]–[6].

In this work, we present a shortest path inverse optimal control method, which is applied to train a predictive model of human motor control. The inverse optimal control method is thereby used to learn the parameters of an optimal control problem from demonstrated state and input trajectories. In particular, it learns both the objective function and constraints

of an underlying infinite-horizon optimal control problem from observed trajectory segments of finite length using optimality conditions of a corresponding shortest path problem and a candidate constraint set. The optimality conditions are derived based on Bellman’s principle of optimality [7] and the Karush-Kuhn-Tucker (KKT) optimality conditions [8]. The proposed method is convex for objective functions that are linear in their parameters and for general nonlinear systems, where relevant constraints are identified from the candidate constraint set using Lagrange multipliers. The method is utilized to train a predictive model of movements of three human subjects from a human manipulation task.

We set up a human manipulation experiment, where three human subjects manipulated one end of a passive kinematic object whose position was changed consecutively by a robot. In this context, the goal of the inverse learning method is to train a predictive model of human movements. The underlying hypothesis is that the demonstrations of the human manipulation task are optimal with respect to an infinite-horizon constrained optimal control problem. The experimental study highlights the potential of the proposed learning approach by providing good predictive performance for individual human movements. In particular, the proposed shortest path formulation is shown to be beneficial for sub-optimal execution, i.e. disregard the reactive human motor control component in the application considered in this paper.

Related inverse optimal control approaches are presented in [9]–[15]. The approaches in [9]–[11] can be interpreted as an inverse method of an infinite-horizon optimal control problem, but they are restricted to unconstrained, linear systems and quadratic objective functions. In [12], a bilevel approach to solve an inverse unconstrained optimal control problem is presented. The techniques closest to our method are [13]–[15], where the KKT conditions are similarly used for learning the stage cost but the constraints are assumed to be known. The two main distinctions of our approach with respect to [13]–[15] are the consideration of an optimal control problem with an infinite horizon and the simultaneous identification of constraints from a candidate constraint set that is constructed from data with a convex optimization problem. By using a shortest path formulation, the required trajectory segment for learning the parameters of the underlying optimal control problem can be shorter, e.g. compared to [14], and the learned parameters are invariant with respect to the chosen trajectory segment. As for the application, the incorporation of constraints results in better predictions of human movement, whereas the consideration of a shortest path formulation allows for isolating trajectory segments where the predictive component is dominant, i.e. where the hypothesis of optimal demonstrations with respect to an optimal controller is valid.

M. Menner, P. Worsnop, and M.N. Zeilinger are with the Institute for Dynamic Systems and Control, ETH Zurich, 8092 Zurich, Switzerland {mmenner, pworsnop, mzeilinger}@ethz.ch

This work was supported by the Swiss National Science Foundation under grant no. PP00P2_157601 / 1.

II. SHORTEST PATH INVERSE OPTIMAL CONTROL

This section presents an inverse optimal control (IOC) approach based on a shortest path formulation to learn an objective function and constraints from observations. The observations are represented as trajectories of state measurements $x(k) \in \mathbb{R}^n$ and inputs $u(k) \in \mathbb{R}^m$ at time-step k , where

$$x(k+1) = f(x(k), u(k)) \quad (1)$$

with the potentially nonlinear function $f(\cdot)$ modeling the evolution of the state. For the derivation of the inverse method in this section, we assume that $f(\cdot)$ is given. Section IV discusses how to identify $f(\cdot)$ for the considered application.

Observed trajectories are assumed to be optimal with respect to an infinite-horizon constrained optimal control problem, i.e. $x(k+i) = x_i^*$ and $u(k+i) = u_i^* \forall i \geq 0$ with

$$\{x_i^*, u_i^*\}_{i=0}^\infty = \arg \min_{x_i, u_i} \sum_{i=0}^\infty l(x_i, u_i; L) \quad (2a)$$

$$\text{s.t. } x_{i+1} = f(x_i, u_i) \quad \forall i \geq 0 \quad (2b)$$

$$C(x_i, u_i) \leq 0 \quad \forall i \geq 0 \quad (2c)$$

$$x_0 = x(k) \quad (2d)$$

with stage cost $l(x_i, u_i; L)$ defined as a parametric function with parameters L , constraint set $C(x_i, u_i) \leq 0$, and initial state $x(k)$. The notation $\{\cdot\}_{i=0}^\infty$ is used to indicate indices from $i = 0$ to ∞ . The goal in this work is to train a predictive model by learning both $l(x_i, u_i; L)$ and $C(x_i, u_i)$ from state and input measurements, which is referred to as the inverse problem to (2) in the following.

Problem Definition: The first difficulty in the inverse problem of (2) is that measurements $x(k), u(k)$ are not available for $k \rightarrow \infty$ but only in some finite segment. We address this using a shortest path formulation (Section II-A). For cases, where the constraint set $C(\cdot, \cdot)$ is unknown, we propose the construction of a candidate constraint set. The main step of the proposed approach is the derivation of optimality conditions of the shortest path formulation using the candidate constraint set (Section II-B). The optimality conditions are then used to simultaneously identify constraints from the candidate set and learn the stage cost parameters.

A. Formulation of infinite-horizon as shortest path problem

We formulate the infinite-horizon problem as a shortest path problem of finite length e and show that the minimizers of both the infinite-horizon problem and the shortest path problem are identical along the path, i.e. from time k to $k+e$. Let $X^m := [x(k)^T \ x(k+1)^T \ \dots \ x(k+e)^T]^T \in \mathbb{R}^{n(e+1)}$ and $U^m := [u(k)^T \ u(k+1)^T \ \dots \ u(k+e-1)^T]^T \in \mathbb{R}^{me}$ be the collection of state and input measurements, respectively, over the time interval k through $k+e$. If X^m, U^m describe the shortest path, then they (at least locally) minimize

$$\begin{aligned} \{X^m, U^m\} = \arg \min_{x_i, u_i} \sum_{i=0}^{e-1} l(x_i, u_i; L) \\ \text{s.t. } x_{i+1} = f(x_i, u_i) \\ C(x_i, u_i) \leq 0 \quad i = 0, \dots, e-1 \\ x_0 = x(k) \\ x_e = x(k+e). \end{aligned} \quad (3)$$

Using Bellman's principle of optimality [7], we can show that X^m, U^m then also correspond to minimizers of (2) for $i = k, \dots, k+e$, which is formally stated in the following theorem.

Theorem 1. *Consider a trajectory segment of measurements X^m, U^m from a dynamical system (1). If the observed inputs U^m are the result of the optimal control problem in (2) for times $k, \dots, k+e-1$, then X^m, U^m also (at least locally) minimize the optimization problem in (3).*

Proof. The optimization problem in (2) can be written as

$$\begin{aligned} J^*(x(k)) = \min_{x_i, u_i} \sum_{i=0}^{e-1} l(x_i, u_i; L) + \sum_{i=e}^\infty l(x_i, u_i; L) \\ \text{s.t. } (2b), (2c), (2d). \end{aligned} \quad (4)$$

If x_e^* is known, then, using Bellman's principle of optimality [7] with $x_e = x_e^*$, (4) can be formulated as

$$\begin{aligned} J^*(x(k)) = \min_{x_i, u_i} \sum_{i=0}^{e-1} l(x_i, u_i; L) + J^*(x_e^*) \\ \text{s.t. } x_{i+1} = f(x_i, u_i) \quad i = 0, \dots, e-1 \\ C(x_i, u_i) \leq 0 \quad i = 0, \dots, e-1 \\ x_0 = x(k) \\ x_e = x_e^*. \end{aligned} \quad (5)$$

Hence, the minimizers of (2) and (5) are equal for all $i = 0, \dots, e$. The result follows with $x_e^* = x(k+e)$. \square

Note that problem (3) differs from a standard finite-horizon formulation as used in [14] by the end-point constraint $x_e = x(k+e)$, which makes a key difference for learning the problem parameters, as will be illustrated in Section III.

Remark 1. *The shortest path formulation originates from the hypothesis that demonstrations are optimal with respect to the infinite-horizon problem in (2). For a different model/hypothesis, the formulation of the inverse problem can differ. A particular advantage of the shortest path formulation is that any path along the measured trajectory can be used for learning. This allows for selecting particular paths where the assumption of optimal execution/data is fulfilled 'more closely', e.g. high signal-to-noise ratio or negligible reactive human motor control component in the application considered.*

B. Optimality conditions

In the following, we derive optimality conditions of the shortest path problem in (3) and show how they can be used for learning both parameters of the stage cost and constraints. First, we express the optimization problem in (3) in terms of the inputs u_i by recursively defining $x_i = F_i(U, x_0)$:

$$F_i(U, x_0) := \begin{cases} x_0 & \text{if } i = 0 \\ f(F_{i-1}(U, x_0), u_{i-1}) & \text{else} \end{cases} \quad (6)$$

with $U := [u_0^T \ u_1^T \ \dots \ u_{e-1}^T]^T$. Hence, the resulting optimization problem is given as

$$\begin{aligned} \min_U \sum_{i=0}^{e-1} l(F_i(U, x(k)), u_i; L) \\ \text{s.t. } C(F_i(U, x(k)), u_i) \leq 0 \quad i = 0, \dots, e-1 \\ F_e(U, x(k)) = x(k+e), \end{aligned} \quad (7)$$

where we use $x_0 = x(k)$. The Lagrangian $\mathcal{L}(U, \lambda, \nu, L)$ of the optimization problem in (7) is given by

$$\begin{aligned} \mathcal{L}(U, \lambda, \nu, L) = & \nu^T (F_e(U, x(k)) - x(k+e)) \\ & + \sum_{i=0}^{e-1} l(F_i(U, x(k)), u_i; L) + \lambda_i^T C(F_i(U, x(k)), u_i) \end{aligned} \quad (8)$$

with Lagrange multipliers $\lambda_i \geq 0$ and $\nu \in \mathbb{R}^n$, cf. [16], and L denoting the parameters of the stage cost $l(x_i, u_i; L)$. Using $\mathcal{L}(\cdot)$ in (8), the KKT optimality conditions for the trajectory segment are given by

$$\nabla_U \mathcal{L}(U, \lambda, \nu, L) = 0 \quad (9a)$$

$$\lambda_i^T C(F_i(U, x(k)), u_i) = 0 \quad i = 0, \dots, e-1 \quad (9b)$$

$$\lambda_i \geq 0 \quad i = 0, \dots, e-1 \quad (9c)$$

$$C(F_i(U, x(k)), u_i) \leq 0 \quad i = 0, \dots, e-1 \quad (9d)$$

$$F_e(U, x(k)) - x(k+e) = 0. \quad (9e)$$

1) *Construction of candidate constraint set:* Eq. (9d) will hold for any observed trajectory with optimal execution (primal feasibility), however, the function C might be unknown. If C is unknown, we propose to use (9d) to construct candidate constraints $\bar{C}(x_i, u_i)$ as the convex hull of all observed data points of the form $P[x_i^T \ u_i^T]^T \leq p$. A subset of the candidate constraints is then identified as constraints via the KKT conditions. A method for computing the convex hull, i.e. P and p , is, e.g., presented in [17].

2) *Optimality conditions for learning:* The idea of the proposed approach is to solve (9) for the parameters L of the stage cost $l(x_i, u_i; L)$ as well as for λ_i and ν , given measurements X^m , U^m and the candidate constraints $\bar{C}(x_i, u_i)$, i.e.

$$\nabla_U \bar{\mathcal{L}}(U, \lambda, \nu, L)|_{U=U^m} = 0 \quad (10a)$$

$$\lambda_i^T \bar{C}(x(k+i), u(k+i)) = 0 \quad i = 0, \dots, e-1 \quad (10b)$$

$$\lambda_i \geq 0 \quad i = 0, \dots, e-1 \quad (10c)$$

with the approximate Lagrangian $\bar{\mathcal{L}}(\cdot)$ defined as in (8) where $\bar{C}(F_i(U, x(k)), u_i)$ replaces $C(F_i(U, x(k)), u_i)$. Eq. (9d) is only needed for the construction of candidate constraints and (9e) holds by construction. Hence, both $\bar{C}(x(i), u(i)) \leq 0$ and (9e) are not needed for learning the stage cost parameters, cf. (9) with (10). The feasibility problem in (10) is convex if $l(x_i, u_i; L)$ is linear in L . One can show that (10) is always feasible using the convex hull as the candidate constraint set, provided optimal and noise-free data.

The Lagrange multipliers λ_i and their values are essential in the proposed IOC approach in order to identify constraints from the candidate set. Each scalar $\lambda_{i,j}$ can be interpreted as a force keeping the optimization problem (7) from violating the corresponding primal constraint $\bar{C}_j(x_i, u_i) \leq 0$ at time i . In other words, the value of a dual variable $\lambda_{i,j}$ indicates the sensitivity of the optimization problem to the corresponding constraint [16]. We define a measure for the identification of constraint j as $\Lambda_j \geq \bar{\Lambda}$ with

$$\Lambda_j = \sum_{i=0}^{e-1} \lambda_{i,j}, \quad (11)$$

where $\bar{\Lambda} \geq 0$ is a problem-specific threshold value. If, e.g., $\Lambda_j = 0$, the j^{th} constraint does not affect the minimizer of

the optimization problem and does not represent a constraint. If, however, the value of Λ_j is very high, the minimizer is strongly affected by the constraint j and the constraint is therefore crucial in explaining the observed trajectory. Hence, Λ_j relates directly to the importance of constraint j . The larger Λ_j , the more important is constraint j . We utilize this relation to identify constraints from the candidate set. The identified constraints are used in the predictive model, along with the learned parameters of the stage cost.

C. Sub-optimal and noisy data

Eq. (10) will be feasible if, and only if, the trajectory is the solution of an optimal control problem of the form (2). In practice, however, even if this modeling assumption is correct, the feasibility problem in (10) will not be satisfied exactly due to measurement or process noise. In order to learn from sub-optimal or noisy data, we propose to solve the relaxed problem

$$\begin{aligned} \min_{L, \nu, \lambda_i} \quad & \|\nabla_U \bar{\mathcal{L}}(U, \lambda, \nu, L)|_{U=U^m}\|_2^2 \\ \text{s.t.} \quad & \lambda_i^T \bar{C}(x(k+i), u(k+i)) = 0 \\ & \lambda_i \geq 0 \quad i = 0, \dots, e-1. \end{aligned} \quad (12)$$

It is easy to verify that $\|\nabla_U \bar{\mathcal{L}}(\cdot)|_{U=U^m}\|_2^2 = 0$ indicates optimality with respect to (10) and that (12) is always feasible.

Remark 2. The use of a shortest path formulation in this work is reflected through the term $\nu^T (F_e(U, x(k)) - x(k+e))$ in (8). Thus, an inverse approach with finite horizon as in [14] is obtained with $\nu = 0$.

Remark 3 (On active and identified constraints). A constraint j is active if $\bar{C}_j(x_i, u_i) = 0$ at time i . Using the proposed method for constructing candidate constraints, there are always active candidate constraints. However, it is important to note that not all active candidates yield $\Lambda_j > 0$; it is also possible that candidate j is active, i.e. $\bar{C}_j(x_i, u_i) = 0$, and $\Lambda_j = 0$. Inversely, $\Lambda_j = 0$ does not mean that the candidate j is never active but that the observed trajectory would have been the same with and without candidate j . Hence, candidate constraint j is not identified as constraint if $\Lambda_j = 0$. Section III illustrates this concept in a simulation example.

III. ILLUSTRATIVE EXAMPLE

In this section, we illustrate the IOC procedure and highlight its key benefits in simulation for a pendulum with the discrete-time state-space representation:

$$\begin{bmatrix} x_1(k+1) \\ x_2(k+1) \end{bmatrix} = \begin{bmatrix} x_1(k) + T_s x_2(k) \\ x_2(k) - T_s \frac{g}{l} \sin x_1(k) \end{bmatrix} + T_s \begin{bmatrix} 0 \\ \frac{1}{ml^2} \end{bmatrix} u(k)$$

with $x_1(k) = \theta(t)$ at $t = kT_s$ and $T_s = 0.01\text{s}$, $g = 9.81\text{m/s}^2$, $l = 1\text{m}$, and $m = 1\text{kg}$. $\theta(t)$ is the angle and $u(t)$ is the applied torque in Nm, where $|u(t)| \leq \bar{u}$ with $\bar{u} = 5\text{Nm}$ is assumed to be the available torque. In the following, we consider an optimal controller of the form (2) with constraints $u_i \leq 5$ and $-u_i \leq 5$ and stage cost $l(x_i, u_i; Q^{\text{gt}}, r^{\text{gt}}) = x_i^T Q^{\text{gt}} x_i + r^{\text{gt}} |u_i| + u_i^2$. The goal in this example is to learn the constraints and the parameters Q^{gt} and r^{gt} .

A. Learning with shortest path and finite horizon methods

First, we highlight main differences between the proposed shortest path formulation and two finite-horizon methods, i.e. a method using the KKT conditions similarly as in [14] and a probabilistic IOC method which uses a likelihood maximization similarly as in [18]. The finite-horizon KKT method differs from the presented approach by virtue of the term $\nu^T(F_e(U, x(k)) - x(k + e))$ in (8) and thus, follows readily with $\nu = 0$ (removing the term). The proposed IOC approach, similarly as the approach in [14], yield a convex semi-definite program, which can e.g. be solved with MOSEK [19], whereas the likelihood maximization method yields a non-convex optimization problem, which in this example is solved with a projected gradient descent method.

Figure 1 shows results with trajectory segments from $t = 0$ s through t_e generated with $Q^{\text{gt}} = I$ and $r^{\text{gt}} = 0$, where we enforce $Q \succeq 0$. The middle plot shows that the proposed method only needs a segment from $t = 0$ s through $t_e \approx 0.5$ s to find the ground truth. Both methods with finite horizon are not able to learn the ground truth even if the segments are long and $\theta(t)$ is close to stationarity, cf. $Q_{12} \approx 1$ at $t_e = 1000$ s.

B. Learning with and without candidate constraints

Next, consider the trajectories with $Q^{\text{gt}} = 10I$ and $r^{\text{gt}} = 1$ for comparing methods with and without candidate constraints using segments from t_i to $t_i + 2$ s, cf. the top plot in Figure 2.

IOC, constrained (2nd plot from the top): The first step is to construct candidate constraints for the input $u(k)$:

$$u(k) \leq g_u \quad (13a)$$

$$-u(k) \leq g_l \quad (13b)$$

where g_u and g_l depend on the chosen segment and are displayed in red (diamond markers) and green (triangle markers), respectively. The algorithm returns Q and r as well as Λ_1 and Λ_2 , which are defined in (11) and correspond to the candidate constraints (13a) and (13b), respectively. The parameters Q and r are very close to the ground truth for all t_i . If $t_i < 0.96$ s, $g_u = 5$ and $\Lambda_1 > 0$ suggesting that $u(k) \leq 5$ is indeed a constraint. If $t_i > 0.96$ s, $g_u < 5$ and $\Lambda_1 = 0$ suggesting that $u(k) \leq g_u < 5$ is not a constraint, which is correct, as the constraint is not active. For all t_i , $g_l < 5$ and $\Lambda_2 = 0$ (not displayed) suggesting that $-u(k) \leq g_l < 5$ is not a constraint. Overall, Q is learned reliably and for $t_i < 0.96$, $u(k) \leq u_{\max}$ is learned as constraint. The trajectory does not provide conclusive evidence about the existence of a lower bound, i.e. $-u(k) \leq u_{\max}$, which is expected as $g_l < 5 \forall t_i$.

IOC, unconstrained (3rd plot from the top): If $t_i > 0.96$ s, Q and r are very close to the ground truth, which is expected since the control problem is virtually unconstrained in these segments. However, if no candidate constraints are constructed a priori, Q and r differ for $t_i < 0.96$ s as the observed trajectory cannot be explained by means of an unconstrained optimal control problem.

Finite-horizon IOC, constrained (bottom plot): The method learns the constraint $u(k) \leq 5$ using similar arguments as SP-IOC, however, it fails to capture the ground truth stage cost parameters with $r \approx 0$ and Q not close to Q^{gt} for all trajectory segments.

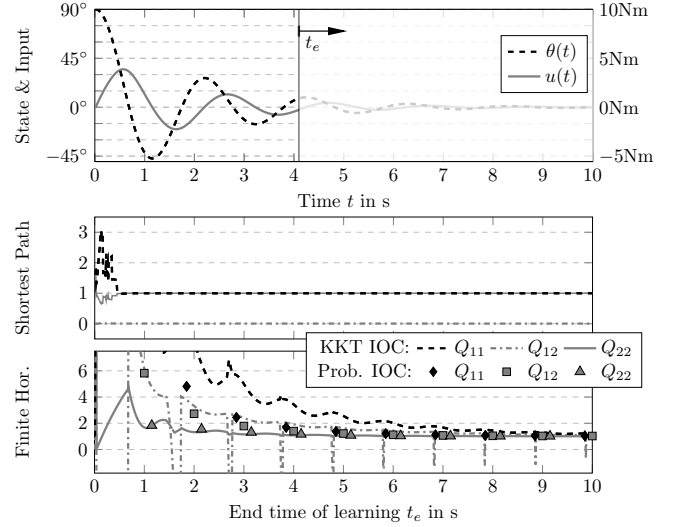


Fig. 1. Top: State and input trajectories. Middle: Q learned with SP-IOC. Bottom: Q learned with two finite-horizon methods: KKT and maximum likelihood.

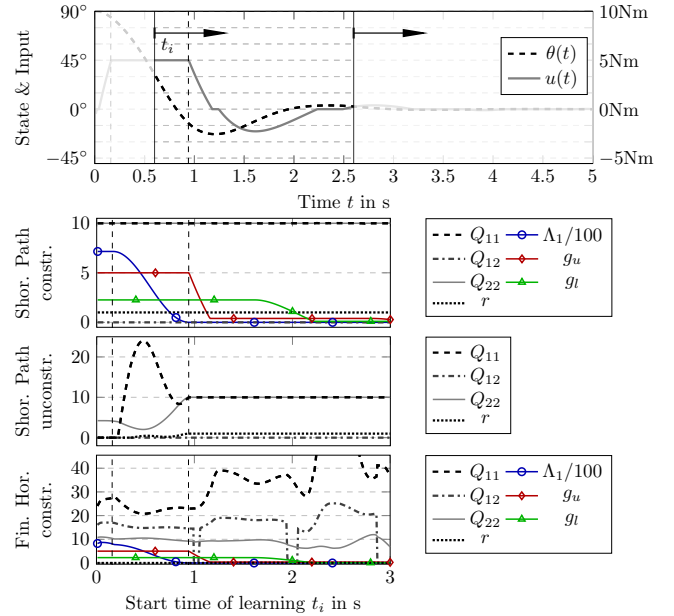


Fig. 2. Top: State and input trajectories. 2nd from the top: Parameters learned with the proposed method with candidate constraints. 3rd from the top: Q learned without candidate constraints. Bottom: Parameters learned with finite-horizon method with candidate constraints.

C. Summary of analysis

In this section, we have illustrated the benefits of the proposed approach. In particular, we showed the candidate constraint construction and how to simultaneously learn parameters of the stage cost and identify constraints from the candidate set. Further, we have shown that the proposed shortest path formulation only requires a short segment of measurements to learn the stage cost parameters and identify constraints, whereas finite-horizon approaches require a comparably long segment. Moreover, we have shown the importance of the candidate constraint set as a substantial component for correctly identifying the stage cost.

IV. MANIPULATION OF A PASSIVE KINEMATIC OBJECT

In this section, we show how to train a predictive model for human movements in a manipulation task using the proposed method. We conducted experiments with three human subjects where the underlying hypothesis is that humans plan their movements by solving a constrained optimal control problem.

A. Experiment description and system modeling

In the experiment, the human subjects manipulated one end of an object whose position was changed consecutively by a robot. The manipulation task was set up to provide a foreseeable environment triggering the human's predictive motor control component such that the reactive control component can be disregarded (at least at the beginning of the movement). The object was articulated and unactuated and was composed of three lightweight wooden links and one cardboard handle, which acted as both a revolute joint and the manipulation point, cf. Figure 3. Hence, it had four revolute joints, one connecting its end link to the robot (joint 1), two connecting the three wooden links (joint 2 & 3), and the cardboard handle (joint 4), which was gripped by the subject such that the forearm and the handle acted as a single rigid body.

After familiarizing themselves with the robot, the human was instructed to achieve specific angles for two of the object's joints, the joint connecting the object to the robot (joint 1 in Figure 3) and the first joint after that (joint 2), both of which have vertical rotational axes (perpendicular to the ground). The target angles were communicated to the subjects visually by reference-markers attached to the links. The subjects were asked to only move when the robot was stationary. First, the robot moved to disturb the system state. When the robot's motion ended, the subject corrected the reference error. Motion capture sensors were placed on all links of each kinematic chain and recorded through the Phasespace Python API.

The derivation of the individual movement model, i.e. the system dynamics, of each subject is based on modeling the passive kinematic object and the human arm as a kinematic chain [20] whose parameters were identified from measurements. In this model, the base frame is attached to the torso and the manipulation frame is attached to the grip location of the hand. Ball joints such as the shoulder joint are modeled as three revolute joints in series with orthogonal axes intersecting at the center of the joint. This leads to the ball joint configuration being described by intrinsic Euler angles rotating around a point in space [21], [22]. The elbow joint is modeled as a single revolute joint. The wrist is modeled as three revolute joints in series, however a wrist brace was used in the experiment to restrict the motions in the frontal and sagittal plane, that is, waving and flapping motions. Pronation and supination (twisting about the forearm) could not be restricted by the brace, however the experiment was designed such that the kinematic chain of the object itself constrained this movement. Both the placement of the motion capture markers and the kinematic modeling are shown in Figure 3.

The system state $x(t) = [x_h(t)^T x_o(t)^T]^T$ is composed of the joint angles of the human, $x_h(t) \in \mathbb{R}^4$, and of the object, $x_o(t) \in \mathbb{R}^4$. The input to the system, $u(t) = \dot{x}_h(t)$, is given

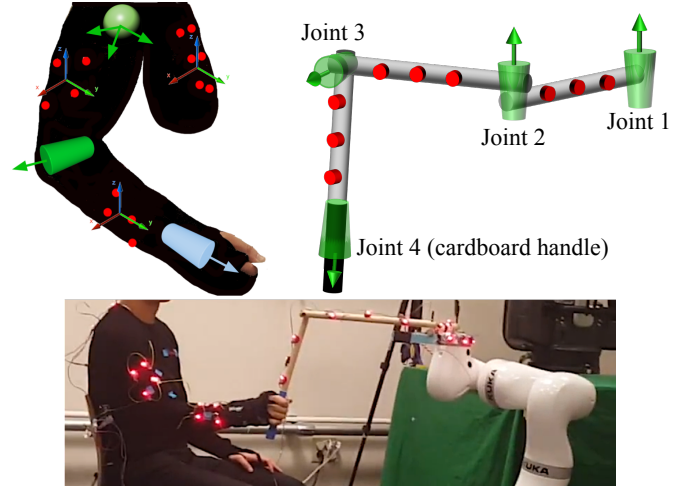


Fig. 3. Top: Modeling of the human arm and the object. Bottom: Experiment setup with the Kuka LBR iiwa robot. Joints included in the model are shown in green, while the blue joint represents a freedom of motion that was constrained by experiment design. The motion capture markers are illustrated in red.

by the joint velocities of the human arm. The velocities of the object joint angles are given by:

$$\dot{x}_o(t) = J_o^\dagger(x_o(t))V_g(t), \quad (14)$$

where $J_o(x_o(t)) \in \mathbb{R}^{6 \times 4}$ is the Jacobian mapping joint velocities of the object to $V_g(t)$, the absolute twist velocity of the manipulation frame, and $J_o^\dagger(x_o(t)) \in \mathbb{R}^{4 \times 6}$ denotes its Moore-Penrose pseudo-inverse [23]. Given that the human maintained a stationary base in the experiment, we can express $V_g(t)$ in terms of the human arm joint velocities and the Jacobian of the human arm, $J_h(x_h(t)) \in \mathbb{R}^{6 \times 4}$:

$$V_g(t) = J_h(x_h(t))\dot{x}_h(t). \quad (15)$$

Using (14) and (15), $\dot{x}_o(t) = J_o^\dagger(x_o(t))J_h(x_h(t))\dot{x}_h(t)$, and thus, the overall dynamics of the system is given by

$$\begin{bmatrix} \dot{x}_h(t) \\ \dot{x}_o(t) \end{bmatrix} = \begin{bmatrix} I \\ J_o^\dagger(x_o(t))J_h(x_h(t)) \end{bmatrix} u(t). \quad (16)$$

In order to obtain the Jacobians, the twists representing the joints in each kinematic chain are identified by recording traces of the subject's range of motion and applying the techniques in [24]. The Jacobians $J_h(x_h(t))$ and $J_o(x_o(t))$ in (16) are derived using the formula for the body Jacobian as in [25].

A discrete-time representation of (16) is derived using an Euler-forward scheme with the sampling time T_s :

$$\begin{bmatrix} x_h(k+1) \\ x_o(k+1) \end{bmatrix} = \begin{bmatrix} x_h(k) \\ x_o(k) \end{bmatrix} + T_s \begin{bmatrix} I \\ J_o^\dagger(x_o(k))J_h(x_h(k)) \end{bmatrix} u(k).$$

An unscented Kalman filter as described in [26] is implemented to estimate the system state, where a static process model is chosen to smoothen the estimated angles, since measurement noise is amplified by the kinematic transformation. The inputs are computed as $u(k) = (x_h(k+1) - x_h(k))/T_s$.

B. Learning predictive model for human movements

Each of the three subjects maneuvered the object 15 times to correct the reference error induced by the robot. For each

experiment, we recorded the entire trajectory from the start of the human movement until the subject was instructed to remain stationary. For reasons discussed in Section IV-C3, we use the initial 1.2s, i.e. $e = 65$ in (10) with sampling time $T_s = 0.0185$ s for learning, which corresponds to roughly 60% of each trajectory. In order to generalize from the available sparse data, we utilize leave-one-out cross-validation [27], where we learn the parameters of the predictive model 15 times, each time removing one of the recorded trajectories. This is done to assess the robustness of the model.

1) *Design choices*: In this work, we train a predictive model with quadratic stage cost. Our goal is to exemplify the proposed method to build a simple predictive model of human movement. Quadratic stage costs are commonly used as objective function in optimal control offering a good compromise between complexity and expressivity, where the cost minimizes a trade-off between tracking a given target and control effort. Note that higher-order or more complex stage cost terms are possible with the proposed framework and there are various possibilities to express human movements [28]. Given that the task requires tracking a reference for only two of the states, we take a stage cost of the form

$$l(x_i, u_i) = (Sx_i - y_s)^T Q (Sx_i - y_s) + u_i^T R u_i,$$

where $y_s \in \mathbb{R}^2$ is the reference, $S = [0_{2 \times 4} \ I_2 \ 0_{2 \times 4}]$ selects the states (two joint angles of the object) tracking y_s , and Q, R are the penalty parameters. We enforce $Q, R \succeq 0$ in order to obtain physically meaningful penalties for both deviation to the target angles and control effort. Also, we restrict the input penalties to $\sum_{i=1}^m R_{ii} = 1$, which fixes the scaling of the stage cost and avoids the trivial solution of all parameters being zero. We train one predictive model without constraints and one with a polytopic candidate constraint set for each subject.

Candidate constraints: The object's states $x_o(k)$ are modeled as unconstrained. The human's states $x_h(k)$ consist of the three shoulder joint angles and the elbow angle; the inputs $u(k)$ are the three angular velocities of the shoulder joint and the angular velocity of the elbow. Constraints on joint angles directly relate to constraints on $x_h(k)$, velocity constraints relate to constraints on $u(k)$, and acceleration constraints are computed as a rate constraint: $a(k) = (u(k+1) - u(k))/T_s$.

2) *Learning results*: Figure 4 shows the mean and standard deviation of Q and R obtained with the proposed IOC method. The most distinct feature is the scale of the parameters Q_{ij} , varying from order 10^{-2} for Subject 1, 10^{-3} for Subject 2, to 10^{-6} for Subject 3. The second most distinct feature is the difference in the diagonal elements of R that reflect movement of the shoulder, i.e. R_{11}, R_{22} , and R_{33} , whereas the penalty on elbow velocity is comparable, i.e. $R_{44} \approx 0.2$ for all subjects. Off-diagonal elements in R are similar across subjects.

Table I shows the sum of Lagrange multipliers as in (11), which are used to identify constraints from the candidate constraint set. The Lagrange multipliers are stated as the mean over all experiments to identify constraints on angle, velocity, and acceleration of shoulder and elbow joints. We consider constraint j as identified if the corresponding Lagrange multiplier $\Lambda_j \geq \bar{\Lambda} = 10^{-3}$. It can be seen that constraints are predominantly on shoulder movement. Constraints on elbow

movement seem less important for all subjects. Note that even though the stage cost parameters in Figure 4 obtained with constrained and unconstrained IOC are relatively close for the individual subject, the resulting prediction models differ by virtue of the constraints identified as in Table I.

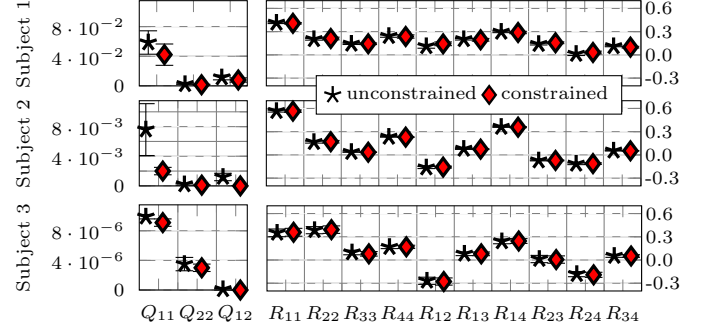


Fig. 4. Mean and standard deviation of cost parameters Q and R for unconstrained learning (black stars) and constrained learning (red diamonds).

TABLE I
LAGRANGE MULTIPLIERS TO IDENTIFY CONSTRAINTS

	Angle		Velocity		Acceleration	
	Shoulder	Elbow	Shoulder	Elbow	Shoulder	Elbow
Subject 1	22.8	0	3.31e-2	0	1.38e-2	8.66e-4
Subject 2	11.5	0	2.78e-1	0	2.15e-2	6.98e-4
Subject 3	3.50	0	4.36e-1	2.86e-4	1.05e-1	0

C. Evaluation of trained human manipulation model

The difficulty in evaluating the quality of the trained model for human-centered experiments is the lack of a ground truth as reference. We therefore assess the quality of modeling human movement as an optimal control problem (2) by comparing the true trajectory with the prediction provided by the model. The predictions are obtained by solving problem (3) with the learned stage cost and identified constraints from the initial position at time $t = 0$ s through $t = t_e = 1.2$ s using IPOPT [29], cf. Figure 5 for a sample prediction. We compute 15 sets of stage cost matrices by leaving out one trajectory for each learning. In order to evaluate the quality of the trained model, we use the left-out measured trajectory for validation against the predicted trajectory, which would result from (3) with the learned stage cost and constraints. This technique ensures that the predicted trajectory is not biased by the corresponding measured trajectory. The mismatch between prediction $\hat{x}_i^j \in \mathbb{R}^8$ and measurement $x^j(i) \in \mathbb{R}^8$ of trajectory j is measured as the root mean square (RMS) error:

$$E^j = \sqrt{\frac{1}{8e} \sum_{i=1}^e \|\hat{x}_i^j - x^j(i)\|_2^2}. \quad (17)$$

1) *Intra-subject evaluation*: First, we compute the errors E^j in (17) for each trajectory j per subject. Figure 5 shows one measured trajectory of Subject 2 and the predictions obtained with the unconstrained and the constrained model. The prediction obtained with the unconstrained model shows a larger RMS error, best seen in the plot of human joint angles.

The prediction obtained with the constrained model shows a lower error. Table II presents the mean and standard deviation over all 15 prediction errors for all subjects. It shows that, generally, the predictions have low errors ($< 3.3^\circ$), where Subject 1 has the lowest ($< 1^\circ$). On average, the presence of constraints improve the predictions by 20%-25%.

TABLE II
PREDICTION ERRORS: UNCONSTRAINED VS. CONSTRAINED

Constraint set	unconstrained	constrained
Subject 1	$0.96^\circ \pm 0.49^\circ$	$0.78^\circ \pm 0.42^\circ$
Subject 2	$3.26^\circ \pm 1.75^\circ$	$2.45^\circ \pm 0.87^\circ$
Subject 3	$1.87^\circ \pm 1.00^\circ$	$1.56^\circ \pm 0.79^\circ$

2) *Inter-subject cross-evaluation*: Next, we analyze the individuality of the trained models, where the error E^j in (17) is computed three times for each trajectory j : We compute the error using the prediction model of the subject who generated trajectory j ; then, we compute E^j of the predicted trajectory \hat{x}_i^j using the other subjects' prediction models, where we use the proposed IOC method with polytopic constraints.

Figure 6 shows an example of a measured trajectory from Subject 1, compared against predictions generated with the models of all subjects. The measured trajectory and the predicted trajectory of Subject 1 are close (error: 0.55°). The predicted trajectories of Subject 2 & 3 show higher errors. Table III states the mean and standard deviation of the errors between measurements of Subject j in columns j and prediction with objective of Subject i in rows i over all trajectories. Hence, a good separation between the subjects means large entries in the off-diagonal entries $i \neq j$. The results show a high confidence in separating Subject 1 from the other two with high confusion errors (3.23° , 2.39° vs. 0.78°). The confidence to identify Subject 2 from a given trajectory is also high with confusion errors (3.99° , 3.59° vs. 2.45°). A less clear separation is observed for Subject 3, where the confusion errors are lower (2.22° , 1.91° vs. 1.56°). Overall, this cross-validation suggests that the models trained to predict the distinct motor behavior are individual.

TABLE III
PREDICTION ERRORS: CROSS-VALIDATION BETWEEN SUBJECTS

Trajectories of	Subject 1	Subject 2	Subject 3
Model			
Subject 1	$0.78^\circ \pm 0.42^\circ$	$3.99^\circ \pm 1.53^\circ$	$2.22^\circ \pm 1.14^\circ$
Subject 2	$3.23^\circ \pm 1.03^\circ$	$2.45^\circ \pm 0.87^\circ$	$1.91^\circ \pm 0.93^\circ$
Subject 3	$2.39^\circ \pm 0.68^\circ$	$3.59^\circ \pm 1.68^\circ$	$1.56^\circ \pm 0.79^\circ$

3) *Benefit of shortest path formulation*: In the following, we discuss the advantages of using a shortest path formulation over a finite horizon in the context of the considered application. If the entire trajectory is used for training and stationarity is reached, i.e. e is large, both the proposed shortest path method and a finite-horizon method are similar. In the context of the considered application, however, we encountered two main challenges when considering the entire trajectory. Firstly, in the final part of the trajectory, the target angles are more or less reached and the measured signals are close to stationarity. As a result, the signal-to-noise ratio is low and can corrupt

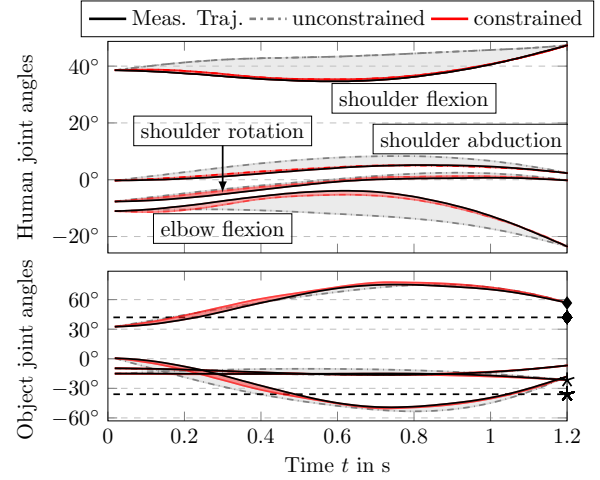


Fig. 5. Measured trajectory in black, predicted trajectory with the unconstrained model in gray (error 4.17°) and the constrained model in red (error 1.40°). The upper plot shows the shoulder flexion, shoulder abduction, and shoulder rotation, as well as elbow flexion. The object states to be tracked are shown in the lower plot as dashed black lines and are related to the corresponding joints with a diamond and a star marker.

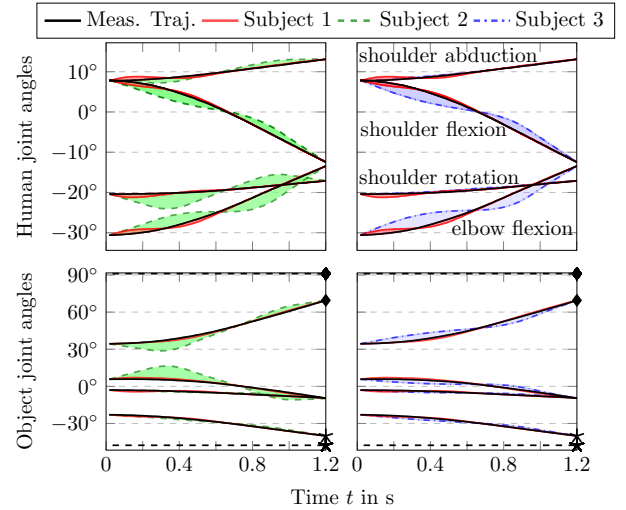


Fig. 6. Measured trajectory of Subject 1 in black, predicted trajectory of Subject 1 in red (error: 0.55°). Left plots: Predicted trajectory of Subject 2 in green (error: 3.62°). Right plots: Predicted trajectory of Subject 3 in blue (error: 1.97°).

learning. Secondly, we observed small corrections around the target angles in the experiment suggesting the presence of reactive movements, which renders the final part of the trajectory not indicative for the predictive human motor control.

For shorter segments, the predictive component dominates both noise and reactive component but the solution from a finite-horizon formulation diverts from that with a shortest path, cf. Section III. The proposed IOC approach allows for using only the initial part of the trajectory for learning where stationarity is not reached. Overall, the presence of both reactive human motor control component and noise do not fulfill the assumption of optimal execution with respect to (2). We used the initial 60% of the trajectory, which was observed to be a good trade-off between segment-length and avoidance of the reactive component.

Figure 7 revisits the trajectory in Figure 5 to illustrate the above discussion on the horizon length e . The upper plot shows the complete recorded trajectory, where some correction around the target angles can be observed for $t \geq 1.4$ s, cf. joint angle marked by the diamond symbol. The lower plot displays the RMS error (17) of the predictions that result from different horizon lengths e . The RMS error increases as a result of both the correction around the target angles and the low signal-to-noise ratio. It highlights that the modeling assumption as an open-loop optimal control problem is suitable for the predictive part, but not in the presence of the reactive component.

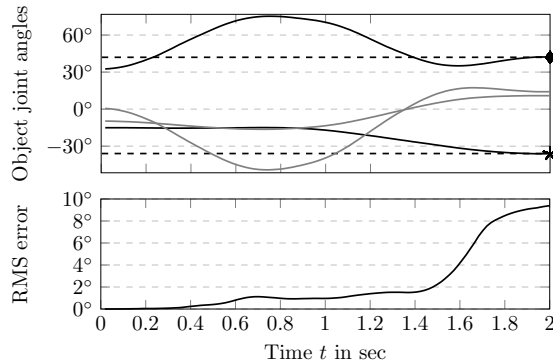


Fig. 7. Top: Target angles to be tracked are shown as dashed black lines and are related to the corresponding joints with a diamond and a star marker. Bottom: RMS error of prediction with different horizon lengths e .

V. CONCLUSION

This paper presented an inverse optimal control approach to learn both cost function parameters and constraints from demonstrations, i.e. state and input measurements of dynamical systems. The shortest path formulation is shown to be the inverse problem to an infinite-horizon optimal control problem. By relying on the Karush-Kuhn-Tucker conditions, the problem is convex for cost functions that are linear in their parameters. We set up a human manipulation experiment to exemplify the proposed approach for modeling and predicting human arm movements. In the experiment, three human subjects manipulated one end of a passive kinematic object whose position was changed consecutively by a robot. The benefits of using a shortest path formulation and the consideration of constraints on human movements were highlighted. The results showed that a model with good predictive capabilities can be learned using a quadratic cost function for both states and inputs together with constraints on shoulder movements using the proposed formulation. Finally, it was shown that the predictive models of the human subjects are individual.

REFERENCES

- [1] J. Lee, P. H. Chang, and D. G. Gweon, "A cost function inspired by human arms movement for a bimanual robotic machining," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2012, pp. 5431–5436.
- [2] D. M. Wolpert, J. Diedrichsen, and J. R. Flanagan, "Principles of sensorimotor learning," *Nature Rev. Neurosci.*, vol. 12, no. 12, p. 739, 2011.
- [3] M. Kawato, "Internal models for motor control and trajectory planning," *Current Opinion in Neurobiology*, vol. 9, no. 6, pp. 718–727, 1999.

- [4] R. S. Johansson and G. Westling, "Coordinated isometric muscle commands adequately and erroneously programmed for the weight during lifting task with precision grip," *Experimental Brain Res.*, vol. 71, no. 1, pp. 59–71, 1988.
- [5] R. S. Johansson and K. J. Cole, "Sensory-motor coordination during grasping and manipulative actions," *Current Opinion in Neurobiology*, vol. 2, no. 6, pp. 815–823, 1992.
- [6] Q. Fu, W. Zhang, and M. Santello, "Anticipatory planning and control of grasp positions and forces for dexterous two-digit manipulation," *J. Neurosci.*, vol. 30, no. 27, pp. 9117–9126, 2010.
- [7] R. E. Bellman, *Dynamic Programming*. Courier Dover Publications, 1957.
- [8] H. W. Kuhn and A. W. Tucker, "Nonlinear programming," in *2nd Berkeley Symposium on Mathematical Statistics and Probability*. University of California Press, 1951.
- [9] R. E. Kalman, "When is a linear control system optimal?" *J. Basic Engineering*, vol. 86, no. 1, pp. 51–60, 1964.
- [10] M. C. Priess, R. Conway, J. Choi, J. M. Popovich, and C. Radcliffe, "Solutions to the inverse LQR problem with application to biological systems analysis," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 2, pp. 770–777, Mar. 2015.
- [11] M. Menner and M. Zeilinger, "Convex formulations and algebraic solutions for linear quadratic inverse optimal control problems," in *Proc. Eur. Control Conf.*, 2018, pp. 2107–2112.
- [12] K. Mombaur, A. Truong, and J.-P. Laumond, "From human to humanoid locomotion - an inverse optimal control approach," *Autonomous Robots*, vol. 28, no. 3, pp. 369–383, 2010.
- [13] A.-S. Puydupin-Jamin, M. Johnson, and T. Bretl, "A convex approach to inverse optimal control and its application to modeling human locomotion," in *Proc. IEEE Int. Conf. Robotics and Automation*, 2012, pp. 531–536.
- [14] P. Englert, N. A. Vien, and M. Toussaint, "Inverse KKT: Learning cost functions of manipulation tasks from demonstrations," *Int. J. Robotics Res.*, vol. 36, no. 13–14, pp. 1474–1488, 2017.
- [15] A. Majumdar, S. Singh, A. Mandlekar, and M. Pavone, "Risk-sensitive inverse reinforcement learning via coherent risk models," in *Robotics: Sci. and Syst.*, 2017.
- [16] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge University Press, 2004.
- [17] C. B. Barber, D. P. Dobkin, and H. Huhdanpaa, "The quickhull algorithm for convex hulls," *ACM Trans. on Math. Software*, vol. 22, no. 4, pp. 469–483, 1996.
- [18] S. Levine and V. Koltun, "Continuous inverse optimal control with locally optimal examples," in *Proc. 29th Int. Conf. Machine Learning*, 2012, pp. 475–482.
- [19] MOSEK ApS, "The MOSEK optimization software," *Online*: <http://www.mosek.com/>, 2010.
- [20] G. Wu, S. Siegler, P. Allard, C. Kirtley, A. Leardini, D. Rosenbaum, M. Whittle, D. D. D'Lima, L. Cristofolini, H. Witte, O. Schmid, and I. Stokes, "ISB recommendation on definitions of joint coordinate system of various joints for the reporting of human joint motion – part i: ankle, hip, and spine," *J. Biomechanics*, vol. 35, no. 4, pp. 543–548, 2002.
- [21] C. Bregler and J. Malik, "Tracking people with twists and exponential maps," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision and Pattern Recognition*, 1998, pp. 8–15.
- [22] G. P. Moll and B. Rosenhahn, "Ball joints for marker-less human motion capture," in *Workshop Appl. Comput. Vision*, 2009, pp. 1–8.
- [23] A. Ben-Israel and T. N. E. Greville, *Generalized inverses: theory and applications*. Springer Science & Business Media, 2003, vol. 15.
- [24] A. M. Bestick, S. A. Burden, G. Willits, N. Naikal, S. S. Sastry, and R. Bajcsy, "Personalized kinematics for human-robot collaborative manipulation," in *Proc. IEEE/RSJ Int. Conf. Intelligent Robots and Syst.*, 2015, pp. 1037–1044.
- [25] R. M. Murray, Z. Li, S. S. Sastry, and S. S. Sastry, *A mathematical introduction to robotic manipulation*. CRC press, 1994.
- [26] E. A. Wan and R. V. D. Merwe, "The unscented kalman filter for nonlinear estimation," in *Proc. IEEE Adaptive Syst. for Signal Process., Commun., and Control Symp.*, 2000, pp. 153–158.
- [27] G. James, D. Witten, T. Hastie, and R. Tibshirani, *An introduction to statistical learning*. New York: Springer, 2013, vol. 112.
- [28] O. Oguz, Z. Zhou, S. Glasauer, and D. Wollherr, "An inverse optimal control approach to explain human arm reaching control based on multiple internal models," *Scientific reports*, vol. 8, no. 1, p. 5583, 2018.
- [29] L. T. Biegler and V. M. Zavala, "Large-scale nonlinear programming using ipopt: An integrating framework for enterprise-wide dynamic optimization," *Comput. & Chemical Eng.*, vol. 33, no. 3, pp. 575–582, 2009.