A New Robotic Knee Impedance Control Parameter Optimization Method Facilitated by Inverse Reinforcement Learning

Wentao Liu, *Student Member, IEEE*, Ruofan Wu, Jennie Si, *Fellow, IEEE*, and He Huang, *Senior Member, IEEE*

***Abstract*—Recent efforts in the design of intelligent controllers for configuring robotic prostheses have demonstrated new possibilities in improving mobility and restoring locomotion for individuals with lower-limb disabilities. In these efforts, personalizing the controller of the robotic device is a crucial step in order to meet individual user's needs and physical conditions. Reinforcement learning (RL) based control designs are among some of the most promising approaches to achieving real-time, optimal adaptive tuning capability. However, such designs to date rely on subjectively determining human-robot walking performance measures, commonly in a quadratic form. To further automate the RL design for robotic knee control parameter tuning and potentially improve human-robot locomotion performance, this study introduces a new bilevel optimization method to objectively specify such control design performance measures via inverse reinforcement learning (IRL), which in turn, will be used in low level (forward) RL design of the impedance control parameters. We demonstrate the effectiveness of the bilevel optimization approach with improved human-robot walking performance using systematic OpenSim simulation studies.**

***Index Terms*—Reinforcement learning, learning from demonstration, wearable robotics, compliance and impedance control.**

I. Introduction

NEW technologies for wearable robotic devices have shown great potential for improving mobility and restoring natural locomotion in individuals with lower limb disabilities [1], [2]. The mechanics (kinetics, kinematics, or impedance) of lower limb joints in these devices usually require modulation by intelligent controllers, tailored to each phase in a gait cycle [3], [4]. However, it has been challenging to design such intelligent controllers in order to meet individual's needs and physical conditions [5], [6].

Manuscript received 24 February 2022; accepted 7 July 2022. Date of publication 27 July 2022; date of current version 23 August 2022. This letter was recommended for publication by Associate Editor M. Huber and Editor J. Kober upon evaluation of the reviewers' comments. This work was supported by the National Science Foundation under Grants 1563454, 1563921, 1808752, 1808898, and 1926998. *(Corresponding authors: He Huang; Jennie Si.)*

Wentao Liu and He Huang are with the UNC/NCSU Department of Biomedical Engineering, North Carolina State University, Raleigh, NC 27695 USA, and also with the University of North Carolina at Chapel Hill, Chapel Hill, NC 27599 USA (e-mail: wliu29@ncsu.edu; hhuang11@ncsu.edu).

Ruofan Wu and Jennie Si are with the School of Electrical, Computer, and Energy Engineering, Arizona State University, Tempe, AZ 85281 USA (e-mail: ruofanwu@asu.edu; si@asu.edu).

Digital Object Identifier 10.1109/LRA.2022.3194326

Some notable progress has been made in recent years towards automating the process of wearable robot personalization. One of the ideas is to estimate the impedance control parameters through model-based methods, such as using a musculoskeletal model [7] or a dynamic model [8]. Another approach is to treat wearable robot personalization as a heuristic human-in-the-loop optimization problem. Several research groups proposed search-based methods to iteratively seek an extremum on the system response surface during walking [9], [10]. This kind of method showed great potential to determine optimal control parameters, such as assistive force, offset timing, or actuation gain. Still, its adaptability and generalizability to changing conditions (e.g., weight change or walking condition change) needs further investigation. Another prevailing solution for robot personalization is data-driven reinforcement learning (RL)-based optimal adaptive control. Such methods are principally scalable and generalizable. They learn directly from data in flexible ways while interacting with the environment [11]–[15].

Central to all these optimization methods is to formulate an appropriate cost function as a representation of human-robot system performance [16]. Also in reinforcement learning, a field strongly connected with optimal adaptive control, it is widely recognized that the cost function provides succinct, robust, and transferable definition of a task, and directly influences the corresponding optimal control law as well as the behaviour of the system [17]. For the problem of wearable robots with human-in-the-loop, controller synthesis often has to answer the following, sometimes related, questions in the design process. First, how to appropriately determine a specific cost function that leads to satisfactory performance in the problem domain? Second, for a complex problem involving multiple performance aspects and the system subject to uncertainty, how to specify a cost function that accounts for multi-attributes of the problem? Furthermore, even when the goal, as reflected by the terms of different performance considerations in a cost function, is relatively clear, how to determine a proper trade-off among those performance aspects? Answering these questions still largely relies on trial-and-error and is usually done by guessing based on knowledge of the domain problem.

Take the robotic knee control problem in our previous works as an example. We formulated the robotic knee parameter personalization as an RL process [12], [14]. To reproduce near-normative joint kinematics, we decomposed a gait cycle into

four phases for control purposes, and adopted peak angle and gait duration timing as the two features reflected in the control objective of each phase. Thus, eight weighting parameters are to be determined in the cost function of a quadratic form. While we successfully demonstrated RL-based controller for automatic tuning of 12 impedance parameters of a robotic knee prosthesis, we selected the eight cost weighting parameters through a trial-and-error process, and settled for a cost function with fixed and uniform weighting factors on the four sets of considered features. In doing so, we simply made an assumption that the weights of the angle error and timing error in the cost function in all four gait phases are identical. In principle, however, this is inconsistent with gait biomechanics. Identical cost functions lead to the same control objectives of knee function across all four gait phases, yet we know that requirements for timing and magnitude of knee motion vary along a gait cycle. Considering terminal stance to mid-swing phase, timing as well as magnitude of knee flexion are both critical to assure foot clearance off the floor [18]. During the terminal swing phase, magnitude of the knee is relatively important because a secure position of knee extension is needed to prepare for weight acceptance [19]. Therefore, there is a clear need for automatically determining such priorities reflected as weightings in the cost function of each gait phase. It can be expected that such cost functions would result in further improved prosthesis control to assist gait.

Inferring an appropriate cost function from examples of desired behaviour has been studied in control system theory and recently in machine learning. Such approaches may be collectively referred to as inverse reinforcement learning (IRL). A seminal work reported in [17], [20] inferred a cost function representation based on measurements of controlled system trajectories or system behaviour. Over the years, IRL has demonstrated its potential in numerous simulated and real-world applications, such as autonomous driving [21], computer graphics [22], and human-robot interaction [23]. IRL therefore provides a feasible approach to determining a cost function as an objective for a controller to optimize. Specifically, it can be used to induce desired behaviour by trading off between multiple confounding performance factors. The potential of IRL for capturing and quantifying a performance objective function in the human-robot system during locomotion was demonstrated in our recent work [24], where we designed an experimental validation procedure to show that IRL was capable of characterizing different human-robot behaviours into mathematical cost functions.

In this work, we propose to develop a new method for robotic knee prosthesis personalization facilitated by IRL. To investigate its potential, and to characterize system response in a wide variety of conditions without potential adverse consequences on human participants and physical devices, we validated the concept using systematic and extensive simulation studies as the initial step. Specifically, we developed a bilevel optimization approach where the low-level RL procedure determines the 12 impedance control parameters, while the high-level IRL procedure provides the RL procedure with four appropriate cost functions in a quadratic form (for the respective four phases in a gait cycle) with 8 corresponding weighting factors.

The main contributions of this study include the following. 1) For the first time, we considered a robotic knee impedance control parameter tuning as a bilevel optimization problem, where IRL is at the high level to provide a quantitative performance objective needed at the low-level RL controller design. 2) We developed an interleaving bilevel learning approach to the design of the 8 weighting factors in the cost function and the design of 12 impedance control parameters. 3) We demonstrated the conceptualization of the new design approach in OpenSim and showed the potential benefit of the IRL-facilitated impedance tuning method. 4) We demonstrated the generalization potential of the bilevel design approach through different walking tasks (i.e., level-ground walking and up-ramp walking).

II. Methods

This study advances our latest reinforcement learning control tuning of robotic knee impedance parameters by employing a more realistic performance objective to be identified by IRL. Previously, we subjectively specified cost functions in the design of control tuning laws. While we demonstrated control performances meeting target kinematic behaviour under those specific cost measures [12], [14], we now hypothesize a control design performance measure that can be determined objectively to reflect human-robot locomotion characteristics is capable to improve the overall walking performance. Therefore, in this study, we aim at developing a generalizable approach to robotic knee impedance control parameter tuning. Towards this goal, we employ the well-established finite state machine impedance control (FSM-IC) framework based on a bipedal walking model in OpemSim [25]. We then discuss how IRL can effectively be used as part of an interleaving process in the design of an RL controller.

A. Finite State Machine Impedance Control of Robotic Knee

Humans reportedly control muscle activities to adjust joint impedance in walking. Compliant behaviours of legs are fundamental to human locomotion [26], [27]. Based on foot–ground interacting events and knee joint movements, a single gait cycle can be decomposed into four consecutive phases: stance flexion (STF), stance extension (STE), swing flexion (SWF) and swing extension (SWE) [12]. We refer to them as Phase 1 through Phase 4 in this report. The control of a robotic knee is built upon FSM-IC to enable continuous walking.

For each of the four gait phases of an FSM (Fig. 1), three impedance parameters are selected to generate a control torque to enable knee motion,

$$I = [K, \beta, \theta_e]^T, \tag{1}$$

where K represents the stiffness, β represents the damping, and θ_e is the equilibrium angle. The device-produced knee joint torque τ used to control knee joint movement is then generated according to the impedance control law

$$\tau = K\left(\theta_e - \theta\right) - \beta\Omega, \tag{2}$$

where θ denotes knee joint angle and Ω represents knee angular velocity. The RL controller adjusts the impedance parameters as

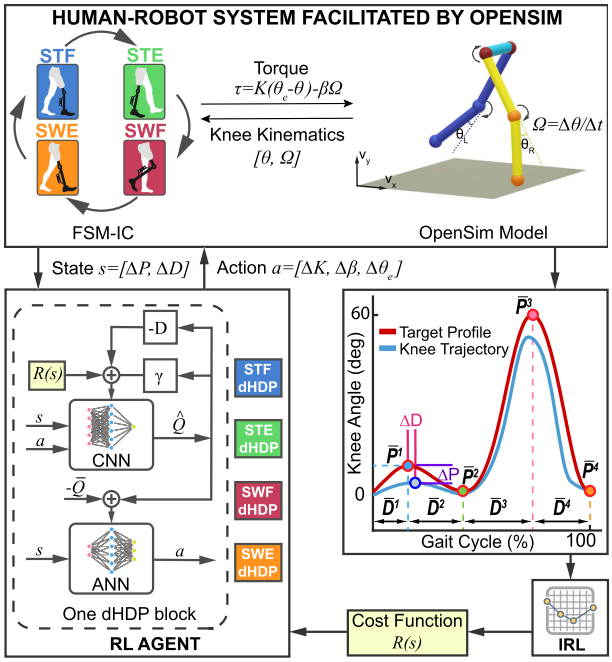

Fig. 1. Schematic of the IRL facilitated impedance tuning for a robotic knee prosthesis within the FSM-IC framework. Top panel: an intrinsic impedance control torque τ is generated from knee kinematic measurements as well as the impedance control parameter settings, which are subject to real-time tuning. Lower left panel: the RL controller makes learning updates at each gait cycle. Its inputs include knee kinematic features, and its outputs include adjustments of the impedance settings. Each phase is associated with one direct heuristic dynamic programming (dHDP) RL controller. Lower right panel: illustration of near-normal knee kinematics (red) and observed knee kinematics (blue). Their respective features in four gait phases are denoted as $\bar{P}^{1\sim4}$ and $\bar{D}^{1\sim4}$, representing the angle and the duration of the respective phase. The phase indices $1 \sim 4$ respectively represent STF, STE, SWF, and SWE. The first phase is used to illustrate how peak error feature ΔP and duration error feature ΔD are formulated. At the end of each learning iteration, the IRL derived cost functions, each has a quadratic form with a total of 8 weighting factors, are used in the RL controller design to obtain the 12 FSM-IC parameters.

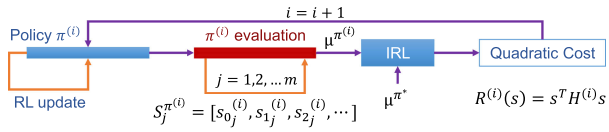

Fig. 2. Schematic of interleaved IRL and RL processes for impedance control parameter tuning. RL impedance control parameter tuning relies on a set of weights $H^{(i)}$ in the quadratic cost (refer to (6)) determined from the IRL procedure. For the i^{th} IRL iteration, the resulting RL control policy $\pi^{(i)}$ is then used in generating a set of m state trajectories $S_j^{\pi^{(i)}}$, which will be used in IRL to identify a new set of weights. The interleaving procedure repeats until meeting convergence criteria.

actions, i.e.,

$$a = [\Delta K, \Delta\beta, \Delta\theta_e]^T \in \mathbb{R}^3. \tag{3}$$

The updated impedance parameters $[K + \Delta K, \quad \beta + \Delta\beta, \theta_e + \Delta\theta_e]^T$ are then applied to FSM-IC to generate knee torque.

B. Robotic Knee Control Tuning for Human Bipedal Walking

We consider robotic knee control tuning a bilevel optimization problem in this study. As shown in Fig. 2, the low level "RL update" loop is within the high level cost function identification loop. For an identified cost function, the RL update is integrated into the FSM-IC framework where RL controllers update their control policy based on measured human-robot system movement (refer to section II-A and Fig. 1). Specifically, each gait phase is associated with an independent RL tuning policy running in parallel. In the following discussion, to avoid notation complication and confusion, we do not specify gait phase in a controller design, which is based on the same principle.

The goal of the automatic tuning approach is to regulate the robotic knee joint to meet a desired knee profile characterized by four discrete target points (Fig. 1). The features of the target knee profile were extracted from normative knee kinematics [28]. Each point in the corresponding phase is associated with two targeted goals, desired peak knee angle $\bar{P}$ and timing $\bar{D}$. We thus define the state variables peak error and duration error by the difference between measured peak knee angle P, duration value D, and their desired targets in every gait cycle

$$s = \left[P - \bar{P}, D - \bar{D}\right]^T = [\Delta P, \Delta D]^T \in \mathbb{R}^2. \tag{4}$$

We assume that there are bounded feature vectors ϕ of the peak error and duration error that can represent system performance and be used as a cost function. For each of the four phases (STF, STE, SWF, SWE in Fig. 1), we let

$$R(s) = \omega^T \cdot \phi(s), \tag{5}$$

where ω quantifies the trade-offs among different performance factors represented by the chosen features of peak error and duration error in this study. Specifically, we assign an efficient and effective cost function structure in each gait phase by employing quadratic features, i.e.,

$$R(s) = s^T H s, \tag{6}$$

where $H = \begin{bmatrix} \omega_1 & 0 \\ 0 & \omega_2 \end{bmatrix} \in \mathbb{R}^{2\times2}$ contains the unknown performance factors to be identified by an IRL procedure.

An RL control policy maps the observed states to impedance control parameters in pursuit of meeting the specified performance objectives. We define an infinite horizon discounted cost-to-go ($\gamma < 1$) for a policy π at the t^{th} gait cycle as

$$V^\pi(s_t) = E\left[R(s_t) + \sum_{\tau=1}^{\infty} \gamma^\tau R(s_{t+\tau}) | \pi\right], \tag{7}$$

and the Q-function as

$$Q^\pi(s_t, a_t) = R(s_t) + \gamma E\left[V^\pi(s_{t+1})\right]. \tag{8}$$

Such a formulation implies that we consider this robotic knee control problem as a discrete time, infinite horizon, discounted problem, and the control design did not require an explicit mathematical description of the human-robot dynamics. Additionally, the Q-function satisfies the Bellman equation

$$Q^\pi(s_t, a_t) = R(s_t) + \gamma Q^\pi(s_{t+1}, a_{t+1}). \tag{9}$$

An optimal policy can be determined from

$$\pi(s_t) \in \underset{a_t \in A}{arg\min} Q^\pi(s_t, a_t). \tag{10}$$

C. Policy Learning in RL-Based Control

To solve the Bellman equation and the control policy from the above, we train a critic neural network and an action neural network as follows.

1) Critic Neural Network (CNN): A CNN is implemented to approximate the Q-function using two layers of weights. Its output value is

$$\hat{Q} = \hat{W}_{c2} g\left(\hat{W}_{c1} z\right), \tag{11}$$

where $\hat{W}_{c1}$ denotes the estimated weights between input and hidden layer, $\hat{W}_{c2}$ is the estimated weights between hidden and output layer. We adopt the hyperbolic tangent $g(\cdot)$ as the activation function. The input z is defined as $[s^T, a^T]^T$. The weights in CNN are updated in order to minimize the critic approximation error, denoted as

$$e_t^c = \left\| \gamma \hat{Q}_t - \left(\hat{Q}_{t-1} - R_{t-1}\right) \right\|. \tag{12}$$

2) Action Neural Network (ANN): An ANN below is a policy network that maps state s to action a for adjusting the impedance parameters in the human-prosthesis system,

$$a = g\left(\hat{W}_{a2} g\left(\hat{W}_{a1} s\right)\right), \tag{13}$$

where $\hat{W}_{a1}$ is the estimated weight vector between input and hidden layers, $\hat{W}_{a2}$ is the weight vector between hidden and output layers. We adopt the same activation function as in CNN. The ANN is designed to backwards pass the prediction error, which is defined as the difference between the approximated Q-function $\hat{Q}$ and desired ultimate objective $\bar{Q}$

$$e_t^a = \left\| \hat{Q}_t - \bar{Q} \right\|. \tag{14}$$

In the equation above $\bar{Q}$ is set as 0 signifying measured feature points meets target profile. Thus the approximated optimal Q-function is $\hat{Q}^*(s,a) = \inf_\pi \hat{Q}^\pi(s,a)$.

The weights of CNN and ANN are updated using a gradient descent backpropagation rule to minimize the respective training errors in the actor and critic networks.

D. Inverse Reinforcement Learning

For the purpose of identifying the unknown weighting factors in (6), an IRL procedure is performed based on observed state trajectories under a control policy. Given a policy π and the resulted state trajectory $S = \{s_0, s_1, s_2, \cdots\}$, we formulate a discounted accumulated feature expectation vector of this trajectory denoted as

$$\mu(\pi) = E\left[\sum_{t=0}^{\infty} \gamma^t \phi(s_t) | \pi\right], \tag{15}$$

where the initial state $s_0 \sim D$, and a feasible set of states and actions are determined from policy π. In this human-prosthesis walking problem formulation, $\mu(\pi) = [\mu_1, \mu_2]^T$ represents feature expectations of ΔP and ΔD. From (5, 7, 15), the value of a policy can be written as

$$V^\pi(s_0) = \omega^T \mu(\pi). \tag{16}$$

As a data-driven approach, we use a sample of m state trajectories (i.e., $j = 1, 2, \ldots, m$) to estimate $\mu(\pi)$ [20] through

$$\mu^\pi = \frac{1}{m} \sum_{j=1}^{m} \sum_{t=0}^{\infty} \gamma^t \phi(s_{tj}). \tag{17}$$

Derived from (6,15,16), the value of a trajectory can be written as

$$V^\pi(s_0) = \sum_{t=0}^{\infty} \gamma^t s_t^T H s_t. \tag{18}$$

The goal of IRL is to find a policy $\widetilde{\pi}$ whose performance is close to the desired performance represented by the state value measure. We thus have

$$\begin{aligned} &\left| E\left[\sum_{t=0}^{\infty} \gamma^t s_t^T H s_t | \pi^*\right] - E\left[\sum_{t=0}^{\infty} \gamma^t s_t^T H s_t | \widetilde{\pi}\right] \right| \\ &= \left| V^{\pi^*}(s_0) - V^{\widetilde{\pi}}(s_0) \right| \\ &= \left| \omega^T \mu(\pi^*) - \omega^T \mu(\widetilde{\pi}) \right| \\ &\leqslant \|\omega\|_2 \|\mu(\pi^*) - \mu(\widetilde{\pi})\|_2 \\ &\leqslant \varepsilon. \end{aligned} \tag{19}$$

To solve for policy $\widetilde{\pi}$ in terms of feature expectations $\mu(\widetilde{\pi})$, the problem can be formulated as

$$\begin{aligned} &\max_{\xi, \omega} \quad \xi \\ &\text{s.t.} \quad V^{\pi^*} \geqslant V^{\pi^{(i)}} + \xi, \; i = 0, 1, 2, \ldots, n; \\ &\qquad \|\omega\|_2 \leqslant 1. \end{aligned} \tag{20}$$

The solved ω places a relative weighting between different performance features. The cost function for control system design is thus determined, which contains 8 unknown factors (2 for each phase) for the four phases of a gait cycle.

Algorithm 1 is a summary of the bilevel, iterative approach to robotic knee impedance parameter tuning. It includes the key steps described above: 1) determining an optimal policy under an IRL specified cost function using RL, 2) evaluation of the obtained control policy by generating several state trajectories for use in IRL, and 3) identification of a cost function (specifying the weighting factors) in the current quadratic cost setting.

III. Implementation

The bilevel design approach to robotic knee impedance control parameter tuning is systematically assessed by simulations using OpenSim, a well-established platform in the field of biomechanics [25]. In the experiments, we implemented Algorithm 1 (also refer to Figs. 1 and 2) in the control of a simulated human-robot system during walking.

A. Human-Robot System Setup

We built a five rigid-segments bipedal model including a pelvis, two thighs, and two shanks on a rigid level platform (refer to Fig. 1) to simulate unilateral above-knee amputee walking, which contains both human-controlled intact joint motion and

Algorithm 1: Bilevel Impedance Control Parameter Optimization Facilitated by IRL.

Initialization (set random initial impedance parameters; select an initially stabilizing policy $\pi^{(0)}$ with random initial weights in CNN and ANN);
Estimate $\mu(\pi^{(0)})$ via sampling using eq. (17);
Set IRL iteration index to $i = 1$;
while IRL stopping criteria are not met **do**
  Determine $H^{(i)}$ according to eq. (16, 18, 20);
  Update cost function $R^{(i)}$ with eq. (6);
  /* Find optimal policy $\pi^{(i)}$ */
  while RL trial stopping criteria are not met **do**
    Obtain state dynamics s;
    if state is out of safety bound **then**
      Reset to initial impedance parameters;
    end
    Obtain action a through $\pi^{(i)}$;
    Obtain instantaneous cost with $R^{(i)}(s)$;
    Update policy $\pi^{(i)}$;
    Update impedance parameters;
  end
  /* Evaluate the current policy */
  for $j \leftarrow 1$ **to** m **do**
    Randomly set initial impedance parameters;
    while RL trial stopping criteria are not met **do**
      Obtain state dynamics s;
      if state is out of safety bound **then**
        Reset to initial impedance parameters;
      end
      Obtain action according to $\pi^{(i)}$;
      Update impedance parameters;
    end
    Store state sequence $S_j^{\pi^{(i)}}$;
    Update evaluation of $\mu(\pi^{(i)})$ using eq. (17);
    Increment j;
  end
  Increment i;
end

robot-controlled prosthetic joint motion. The pelvis segment is linked to the ground platform using a slider joint, which allows the body to move relative to the platform. The thigh segments are linked to the pelvis, and shank segments are attached to the thighs, both by one-degree-of-freedom pin joints. The left knee is defined as intact while the right is prosthetic to enable locomotion. We defined human-controlled joints with prescribed motion according to a well-established, normative data set [29], while the prosthetic knee impedance parameters were updated based on FSM-IC. The simulation was initialized to trivial angles for both knees near stance position. Model settings such as body mass, segment length, and inertial parameters, were specified according to OpenSim lower-limb model [29].

B. Experimental Conditions and Hyperparameters

The goal of the automatic tuning approach is to make the knee kinematics meet the desired near-normal profile. In consideration of walking variability in human locomotion, measurement noise, and other uncertainties, we established tolerance levels of the state variables as shown in the bottom row of Table I. We also set a safety bound based on realistic conditions of balanced walking, which is set at 1.5 times the standard deviations of the respective knee kinematic peak values observed in each phase [28]. If errors exceed the safety bound, which means the subjects may step into unsafe regions with current profile, the impedance parameters will be reset to initial values. Convergence of tuning within a phase is achieved if states remain within the tolerance range for 8 out of 10 consecutive impedance updates. If all four phases converged after tuning, it meets the stopping criteria, and a tuning trial is considered successful. Another stopping criterion is for failed trials when tuning has reached a maximum number of allowed gait cycles (specified as 200).

TABLE I
ANGLE (DEG) AND DURATION (%) BOUNDS

	Phase 1	Phase 2	Phase 3	Phase 4
Safety bound	[10.5, 12]	[7.5, 12]	[9, 12]	[6, 12]
Tolerance bound	[1.5, 2]	[1.5, 2]	[1.5, 2]	[1.5, 2]

The stopping criteria for IRL are as follows: 1) The margin ξ between the target goal and the current performance is $\xi \leqslant 2$; and 2) ξ is non-increasing over the number of IRL iterative procedures. Nonlinear programming was adopted to solve the optimization problem in (20). In a direct heuristic dynamic programming (dHDP) RL controller, the CNN has a three-layered 5-6-1 structure, and ANN with a 2-6-3 structure. Weights in both networks were initialized to random small numbers. We set the discount factor $\gamma = 0.99$, and the neural network learning rate as 0.1.

C. Baseline Methods for Comparison

To provide a baseline approach for comparison with our bilevel impedance control parameter tuning method, we used a manual selection procedure guided by our previous experience in specifying the weighting factors in (6). First, we specified a practical range for the weighting factors based on our extensive previous experience. Second, we used the same set of weighting factors for all four phases to avoid exponential growth in evaluation cases as in our previous works. We aim at providing evidence for the following considerations that motivated this study. 1) As we are already aware that knee biomechanics for each phase is different, how are the controllers in the four respective gait phases fair if they are designed using the same cost function in comparison to the proposed bilevel design? 2) Can we find an appropriate trade-off for the weighting factors in the cost function and how to further optimize system performance without manually going through a tedious or even prohibitive trial-and-error process? 3) Can we expect to see better performance in the bilevel-designed controllers than those developed by manually specified cost functions?

Specifically in comparison studies, the baseline methods were implemented as follows. The controllers for the four phases

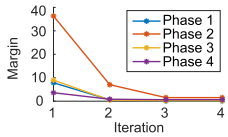

Fig. 3. Margin of feature expectations between desired behaviour μ^* and learned policy μ denoted by $\|\mu(\pi^*) - \mu(\widetilde{\pi})\|_2$ during a representative IRL procedure.

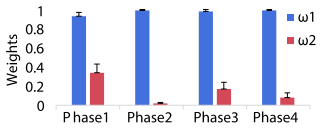

Fig. 4. Weights of peak error and duration error (ω_1 and ω_2) from 30 IRL procedures. Mean values and standard deviations from the mean are shown.

have an identical quadratic form but with manually specified weighting factors. We systematically varied the weighting factor ratio $r = \omega_1/\omega_2$ to cover a wide range of values from 0.1 to 19. We then compared each design from the baseline with the bilevel design developed herein.

D. Performance Evaluations

Systematic evaluations were conducted to assess the performance of bilevel impedance control parameter tuning method. We used the following metrics: 1) the number of IRL iterations to measure the convergence speed of the weighting factor identification procedure; 2) time efficiency of an RL impedance control parameter tuning by the total number of impedance updates; 3) RMSE between measured knee kinematic profiles (robotic knee angles) and target profile.

In the simulations, a trial is a complete tuning process of the RL controller given the cost function determined at the i^{th} iteration of the IRL. Bilevel optimization procedure, as shown in Algorithm 1, was performed 30 times (corresponding to 30 different sets of initial impedance parameters) to obtain statistical results in this study. In each evaluation procedure, a set of $m = 10$ sampled state trajectories were used to induce weighting factors in the quadratic cost function. Therefore, a total of 300 impedance tuning trials were included in the bilevel optimization evaluation. We evaluated the bilevel approach to robotic knee control using level-ground walking and up-ramp walking with slopes of 3° and 5°.

IV. Results and Analysis

For reporting IRL convergence speed, we recorded the margin of feature expectations $\|\mu(\pi^*) - \mu(\widetilde{\pi})\|_2$. Since similar convergence behaviour was observed for all IRL procedures, we demonstrate a representative result in Fig. 3. On average, it took 4.1 iterations for IRL procedure to converge. The IRL configured weights for the cost function are demonstrated in Fig. 4 as a result of 30 bilevel optimization procedures. We noted the average of the ratios between the two weights $r = \omega_1/\omega_2$ for phase $1 \sim 4$ as $2.7, 49.5, 5.8, 12.4$, respectively.

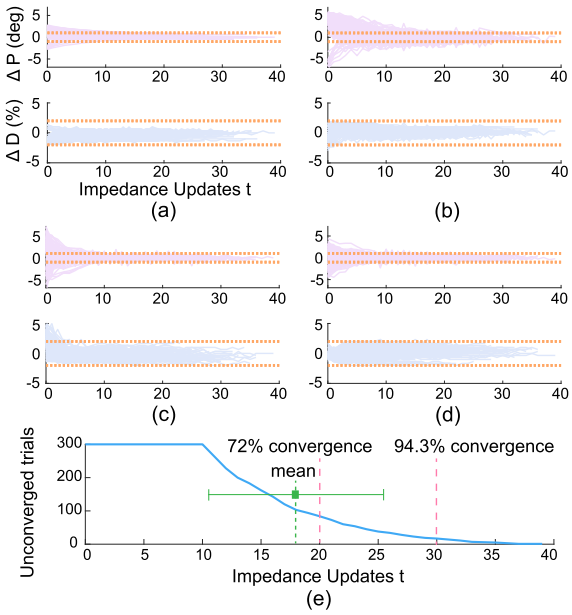

Fig. 5. (a)–(d) Peak error and duration error of phase $1 \sim 4$, respectively in 300 evaluation trials of bilevel optimization, each with different random initial impedance parameters. The horizontal orange dashed lines represent tolerance bounds (refer to Table I). (e) Trial convergence profile: 72% of the trials converged within 20 updates; 94.3% within 30 updates. The average convergence speed is 18 ± 7.7 updates.

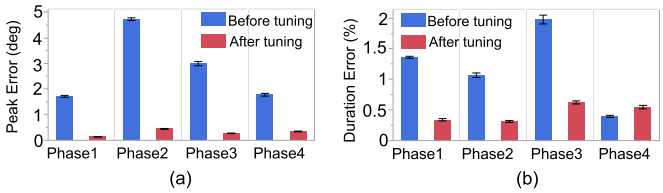

Fig. 6. Performance over 300 trials with one standard deviation from the mean under the condition of before and after tuning: (a) peak angle error and (b) duration error. Before tuning policy is the initial policy while the after tuning policy is the one from the last iteration of each bilevel procedure.

To report time efficiency in impedance tuning, we observed peak error and duration error features during all tuning procedures, each policy was obtained under an IRL determined cost function. As shown in Fig. 5(a)–(d), feature errors in all four gait phases converged within the tolerance bounds, and eventually remained within the bounds. It took an average of 18.7 ± 7.7 gait cycles for the control parameters matching the target behaviour (Fig. 5(e)).

To demonstrate the effectiveness of the bilevel optimization design, we compared the RMSE of the robotic knee angle between the measured and the target profile under two conditions: 1) pre-tuning based on the initial policy, and 2) post-tuning with an IRL facilitated policy. The results were averaged over 300 testing sessions (Fig. 6). All peak error features decreased after tuning with bilevel optimization. The duration error features decreased in phase $1 \sim 3$, but increased from 0.4% to 0.5% in phase 4. This is not surprising as 1) the RL control design allows for a larger tolerance bound of the duration error feature than that of the peak angle error feature, and 2) the peak error

TABLE II
COMPARISON OF PROSTHETIC KNEE CONTROL LEARNING PERFORMANCE DURING LEVEL-GROUND AND UP-RAMP WALKING

	Iteration 1		Iteration 2		Iteration 3	
	Tuning Speed (steps)	Success Rate	Tuning Speed (steps)	Success Rate	Tuning Speed (steps)	Success Rate
level ground	200 ± 0	0.00%	21.6 ± 15.4	100.00%	15.4 ± 5.9	100.00%
up-ramp (3°)	200 ± 0	0.00%	22.0 ± 5.2	99.33%	17.2 ± 5.4	99.33%
up-ramp (5°)	200 ± 0	0.00%	39.8 ± 15.6	87.67%	26.6 ± 10.8	96.33%

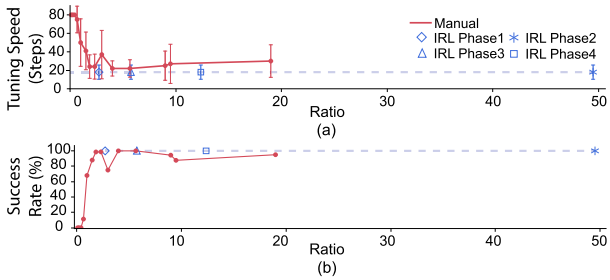

Fig. 7. Comparison of impedance tuning speed and success rate between manually specified cost functions with different ω_1 and ω_2 ratios versus bilevel tuning which automatically induce ω_1 and ω_2 from observed state trajectories. (a) The number of impedance updates and (b) convergence rate.

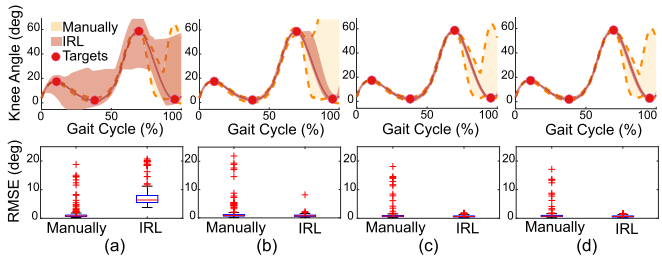

Fig. 8. Comparison of prosthetic knee control performance by evaluating prosthetic side knee kinematics generated by policies from manually specified cost functions (orange) versus IRL induced (salmon). The target knee profile is shown in red with dots representing target features. (a)-(d): observed knee profiles (top row) and RMSEs from the target profile (bottom row) of the first four iterations from both methods.

term is a dominant feature in the cost function as reflected by the learned weights.

Next, we compared the bilevel control parameter optimization performance with baseline methods. Fig. 7 compares controller performance between the bilevel design and the controller designed by manually specified cost functions. By varying ω_1 and ω_2 in H matrix from (6), we selected 14 representative cost functions from the candidate pool. All controllers were trained with 10 randomly initialized impedance parameters to encourage sufficient exploration in policy space. We adopted convergence speed and success rate as performance metrics. Among all the results from manually specified cost functions, the policy associated with a cost function of $\omega_1 = 0.8, \omega_2 = 0.2$ (i.e., with a ratio of 4) outperformed all other specifications of weighting. The mean convergence speed using this set of optimal manually selected weights is 22 ± 8.2 with a 100% success rate. In comparison, the bilevel optimization design (refer to Fig. 5 and 7) has a mean convergence speed and success rate of 18 ± 7.7 and 100% separately. The bilevel design outperformed the best manual specification design by 22.2% in terms of convergence speed.

To further illustrate the respective training process, we compared the performance of two policies: the best policy from manually specified cost functions, and an IRL induced policy with average performance. We performed an evaluation at the end of every iteration, each with 300 trials from a randomly initialized impedance parameter setting. Fig. 8 demonstrates the first four iterations of the prosthetic knee kinematics and its RMSE, under two respective conditions of manual versus IRL induced cost functions. As shown, at the 3rd IRL iteration, bilevel designed controller already reached an acceptable performance range. Under equivalent condition, the manual selection based controller design reached an acceptable performance range in 6 iterations. This indicates that IRL has the potential to efficiently determine an appropriate cost function with improved controller performance.

To evaluate the generalizability of the proposed bilevel design, we trained controllers for up-ramp walking task at two different incline angles (3 degrees and 5 degrees, respectively). Table II is a summary of learning performance measured by tuning speed and success rate during the first three iterations under different walking conditions. Typically it took more steps for the tuning process to converge for walking on a steeper ramp. As training continues, all evaluated walking tasks achieved a similar success rate (all above 95%) at the 3rd IRL iteration.

V. CONCLUSION AND DISCUSSION

This study introduced a new robotic knee impedance control parameter tuning method based on bilevel optimization. In this interleaving bilevel learning approach, the high level IRL procedure aims to automatically characterize an appropriate cost function in a quadratic form, to enable the low level impedance parameter optimization build upon RL. We investigated the potential of this new impedance control parameter tuning method facilitated by IRL for personalizing robotic knee impedance control to enable near-normal knee kinematics during walking. We developed a simulation model in OpenSim for performance evaluation, and validated this approach through extensive simulation studies based on this human-robot system. We also compared the proposed approach with the baseline method, in which RL controllers were designed by manually chosen cost functions under the same environmental conditions.

Experimental results show that both methods were able to tune robotic knee prosthesis control parameters. But the two methods resulted in different control policies as they rely on different approaches to determining the cost function to be used in the controller design. As such, they exhibited different

performances in terms of converging speed and learning success rate. We systematically specified 14 different cost functions manually and assessed the respective controller performance to serve as baseline methods. In comparison, the bilevel design approach does not require a trial-and-error process to specify a cost function, and it is generally inexpensive, i.e., requires only a small number of iterations (4.1 on average), to automatically specify an appropriate cost function. Among all the cases that we tested, the bilevel optimization achieved the best overall performance.

The results also validated our assumption that since knee biomechanics varies in different gait phases during a gait cycle, controller design should be tailored by independent cost functions in different phases. This was shown to enable improved control performance. As such, the bilevel design could be a powerful tool to automate the cost function specification, and thus to further automate the impedance control design and to reduce subjectivity in this process.

Towards further automating the personalization process of impedance parameter tuning, we believe it can be made one step closer by replacing the normative knee profile with alternatives such as the intact knee profile based on successful demonstrations in simulations and in experiments [30], [31]. To test this proposed framework on physical devices would also be an important next step to validate the proposed bilevel design, and more specifically, the designed controllers. Another interesting direction would be to generalize this framework to other devices, such as exoskeletons. Towards these goals, it requires significant research and careful engineering considerations, such as the proper problem formulation, user safety and time efficiency, human gait variations, as well as co-adaptation between human and robot. Based on our previous works, we expect this new framework can be verified by human experiments in the future.

REFERENCES

[1] H. H. Huang, J. Si, A. Brandt, and M. Li, "Taking both sides: Seeking symbiosis between intelligent prostheses and human motor control during locomotion," *Curr. Opin. Biomed. Eng.*, vol. 20, 2021, Art. no. 100314.

[2] W. Huo, S. Mohammed, J. C. Moreno, and Y. Amirat, "Lower limb wearable robots for assistance and rehabilitation: A state of the art," *IEEE Syst. J.*, vol. 10, no. 3, pp. 1068–1081, Sep. 2016.

[3] M. R. Tucker et al., "Control strategies for active lower extremity prosthetics and orthotics: A review," *J. Neuroeng. Rehabil.*, vol. 12, no. 1, pp. 1–30, 2015.

[4] A. H. Shultz, B. E. Lawson, and M. Goldfarb, "Running with a powered knee and ankle prosthesis," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 23, no. 3, pp. 403–412, May 2015.

[5] H. Huang, D. L. Crouch, M. Liu, G. S. Sawicki, and D. Wang, "A cyber expert system for auto-tuning powered prosthesis impedance control parameters," *Ann. Biomed. Eng.*, vol. 44, no. 5, pp. 1613–1624, 2016.

[6] E. J. Rouse, L. M. Mooney, and H. M. Herr, "Clutchable series-elastic actuator: Implications for prosthetic knee design," *Int. J. Robot. Res.*, vol. 33, no. 13, pp. 1611–1625, 2014.

[7] M. F. Eilenberg, H. Geyer, and H. Herr, "Control of a powered ankle–foot prosthesis based on a neuromuscular model," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 18, no. 2, pp. 164–173, Apr. 2010.

[8] E. J. Rouse, L. J. Hargrove, E. J. Perreault, and T. A. Kuiken, "Estimation of human ankle impedance during the stance phase of walking," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 22, no. 4, pp. 870–878, Jul. 2014.

[9] J. R. Koller, D. A. Jacobs, D. P. Ferris, and C. D. Remy, "Learning to walk with an adaptive gain proportional myoelectric controller for a robotic ankle exoskeleton," *J. Neuroeng. Rehabil.*, vol. 12, no. 1, pp. 1–14, 2015.

[10] J. Zhang et al., "Human-in-the-loop optimization of exoskeleton assistance during walking," *Science*, vol. 356, no. 6344, pp. 1280–1284, 2017.

[11] Y. Wen, J. Si, X. Gao, S. Huang, and H. H. Huang, "A new powered lower limb prosthesis control framework based on adaptive dynamic programming," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 9, pp. 2215–2220, Sep. 2017.

[12] Y. Wen, J. Si, A. Brandt, X. Gao, and H. H. Huang, "Online reinforcement learning control for the personalization of a robotic knee prosthesis," *IEEE Trans. Cybern.*, vol. 50, no. 6, pp. 2346–2356, Jun. 2020.

[13] Z. Peng et al., "Data-driven reinforcement learning for walking assistance control of a lower limb exoskeleton with hemiplegic patients," in *Proc. IEEE Int. Conf. Robot. Automat.*, 2020, pp. 9065–9071.

[14] M. Li, Y. Wen, X. Gao, J. Si, and H. Huang, "Toward expedited impedance tuning of a robotic prosthesis for personalized gait assistance by reinforcement learning control," *IEEE Trans. Robot.*, vol. 38, no. 1, pp. 407–420, Feb. 2022.

[15] X. Gao, J. Si, Y. Wen, M. Li, and H. Huang, "Reinforcement learning control of robotic knee with human-in-the-loop by flexible policy iteration," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2021.3071727.

[16] E. Todorov, "Optimal control theory," in *Bayesian Brain: Probabilistic Approaches to Neural Coding*, D. Kenji et al., Eds., Cambridge, MA, USA: MIT Press, Aug. 2013, doi: 10.7551/mitpress/9780262042383.003.0012.

[17] A. Y. Ng and S. J. Russell, "Algorithms for inverse reinforcement learning," in *Proc. Int. Conf. Mach. Learn.*, 2000, vol. 1, pp. 663–670.

[18] J. W. Sensinger, N. Intawachirarat, and S. A. Gard, "Contribution of prosthetic knee and ankle mechanisms to swing-phase foot clearance," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 21, no. 1, pp. 74–80, Jan. 2013.

[19] M. J. Perry, *Gait Analysis: Normal and Pathological Function*. New York, NJ, USA: SLACK, 2010.

[20] P. Abbeel and A. Y. Ng, "Apprenticeship learning via inverse reinforcement learning," in *Proc. 21st Int. Conf. Mach. Learn.*, 2004, pp. 1–8.

[21] S. Levine, Z. Popovic, and V. Koltun, "Nonlinear inverse reinforcement learning with gaussian processes," *Adv. Neural Inf. Process. Syst.*, vol. 24, pp. 19–27, 2011.

[22] S. J. Lee and Z. Popović, "Learning behavior styles with inverse reinforcement learning," *ACM Trans. Graph.*, vol. 29, no. 4, pp. 1–7, 2010.

[23] A. F. Daniele, M. Bansal, and M. R. Walter, "Navigational instruction generation as inverse reinforcement learning with neural machine translation," in *Proc. 12th ACM/IEEE Int. Conf. Hum. Robot Interact.*, 2017, pp. 109–118.

[24] W. Liu, J. Zhong, R. Wu, B. L. Fylstra, J. Si, and H. H. Huang, "Inferring human-robot performance objectives during locomotion using inverse reinforcement learning and inverse optimal control," *IEEE Robot. Automat. Lett.*, vol. 7, no. 2, pp. 2549–2556, Apr. 2022.

[25] S. L. Delp et al., "Opensim: Open-source software to create and analyze dynamic simulations of movement," *IEEE Trans. Biomed. Eng.*, vol. 54, no. 11, pp. 1940–1950, Nov. 2007.

[26] N. Hogan, "Impedance control: An approach to manipulation," in *Proc. Amer. Control Conf.*, 1984, pp. 304–313.

[27] K. Shamaei, G. S. Sawicki, and A. M. Dollar, "Estimation of quasi-stiffness of the human knee in the stance phase of walking," *PLoS One*, vol. 8, no. 3, 2013, Art. no. e59993.

[28] M. P. Kadaba, H. Ramakrishnan, and M. Wootten, "Measurement of lower extremity kinematics during level walking," *J. Orthop. Res.*, vol. 8, no. 3, pp. 383–392, 1990.

[29] D. A. Jacobs, "From the ground up: Building a passive dynamic walker model," 2014. [Online]. Available: https://simtk-confluence.stanford.edu:8443/display/OpenSim33/From+the+Ground+Up%3A+Building+a+Passive+Dynamic+Walker+Model

[30] R. Wu, Z. Yao, J. Si, and H. H. Huang, "Robotic knee tracking control to mimic the intact human knee profile based on actor-critic reinforcement learning," *IEEE/CAA J. Automatica Sinica*, vol. 9, no. 1, pp. 19–30, Jan. 2022.

[31] R. Wu, M. Li, Z. Yao, W. Liu, J. Si, and H. H. Huang, "Reinforcement learning impedance control of a robotic prosthesis to coordinate with human intact knee motion," *IEEE Robot. Automat. Lett.*, vol. 7, no. 3, pp. 7014–7020, Jul. 2022, doi: 10.1109/LRA.2022.3179420.