2019 International Conference on Robotics and Automation (ICRA)
Palais des congres de Montreal, Montreal, Canada, May 20-24, 2019

Offline Policy Iteration Based Reinforcement Learning Controller for Online Robotic Knee Prosthesis Parameter Tuning

Minhan Li, Xiang Gao, Yue Wen, Jennie Si, *Fellow, IEEE,* and He (Helen) Huang, *Senior Member, IEEE*

***Abstract*— This paper aims to develop an optimal controller that can automatically provide personalized control of robotic knee prosthesis in order to best support gait of individual prosthesis wearers. We introduced a new reinforcement learning (RL) controller for this purpose based on the promising ability of RL controllers to solve optimal control problems through interactions with the environment without requiring an explicit system model. However, collecting data from a human-prosthesis system is expensive and thus the design of a RL controller has to take into account data and time efficiency. We therefore propose an offline policy iteration based reinforcement learning approach. Our solution is built on the finite state machine (FSM) impedance control framework, which is the most used prosthesis control method in commercial and prototypic robotic prosthesis. Under such a framework, we designed an approximate policy iteration algorithm to devise impedance parameter update rules for 12 prosthesis control parameters in order to meet individual users' needs. The goal of the reinforcement learning-based control was to reproduce near-normal knee kinematics during gait. We tested the RL controller obtained from offline learning in real time experiment involving the same able-bodied human subject wearing a robotic lower limb prosthesis. Our results showed that the RL control resulted in good convergent behavior in kinematic states, and the offline learning control policy successfully adjusted the prosthesis control parameters to produce near-normal knee kinematics in 10 updates of the impedance control parameters.**

I. Introduction

The robotic prosthesis industry has experienced rapid advances in the past decade. Compared to passive devices, robotic prostheses provide active power to effciently assist gait in lower limb amptuees. Such active devices are potentially beneficial to amputees by providing the capability of decreased metabolic consumption during walking [1], [2], improved performance while walking on various terrains [3], [4], enhanced balance and stability [5], and improved adaptability to different walking speed [6]. In term of control for robotic prostheses, although several ideas [7], [8] have been proposed in recent years, the most commonly used approach in commercial and prototypic devices is still the finite state machine (FSM) impedance control [9]–[11].

The FSM impedance control framework requires customization of several impedance parameters for individual users in order to accommodate different physical conditions. This requirement currently poses a major challenge for broad adoption of the powered prosthesis devices because of the following reasons. For robotic knee prosthesis, the number of parameters to be configured is up to 15 [11], [12]. However, in clinical practice, only 2-3 parameters are practically feasible to be customized by prosthetists manually and heuristically. This procedure is time and labor intensive. Researchers have attempted alternative ways to manual tuning. To mimic the impedance nature of biological joint, intact leg models were studied to estimate the impedance parameters for the prosthetic knee joint [13]–[15]. Yet, the accuracy of these models have not been validated. Our group developed a cyber expert system approach to finding the impedance parameters [16]. This method is promising because of its model-free nature, however, its high demands for knowledge of experienced prosthesis tuning experts impedes its application in the real world. Most recently, some studies proposed to take into account the human's feedback in the optimization for the parameter configuration and demonstrated the promise. However, these methods still have some limitations, such as hard to extend for configuring high dimensional parameters [17] or imposing a prerequisite on the dataset which has to cover all users' preference [18].

In fact, the process of configuring impedance parameters can be formulated as a control problem of solving optimal sequential decisions. Because of the ability to autonomously learn an optimal behavior through interactions rather than explicitly formulate a detailed solution to a specific problem, the reinforcement learning (RL) based control design becomes a natural candidate when it comes to addressing the aforementioned challenges of configuring robotic knee prosthesis to meet individual needs. Recently, RL was successfully applied to solving robotic problems that involve sophisticated and hard-to-engineer behaviors. In most of these successful applications, policy search methods were at the center of the development [19]–[24]. For example, Gu [23] developed an off-policy deep Q-function based RL algorithm to learn complex 7 DoF robotic arm manipulation policies from scratch for a door opening task. Vogt [25] presented a data-driven imitation learning system for learning human-robot interactions from human-human demonstrations. However, deep RL based methods may not be appropriate in some biomedical applications such as the human-prosthesis control problem under consideration. One primary reason is that training data involving human subjects are usually not easily acquired or expensive to collect. Additionally, experimental session involving human subjects

*This work was partly supported by National Science Foundation #1563454, #1563921, #1808752 and #1808898. *(Minhan Li and Xiang Gao are co-first authors. Corresponding authors: He (Helen) Huang; Jennie Si.)*

M. Li, Y. Wen, and H. Huang are with the NCSU/UNC Department of Biomedical Engineering, NC State University, Raleigh, NC, 27695-7115; University of North Carolina at Chapel Hill, Chapel Hill, NC 27599 USA.

X. Gao and J. Si are with the Department of Electrical, Computer, and Energy Engineering, Arizona State University, Tempe, AZ, 85281 USA.

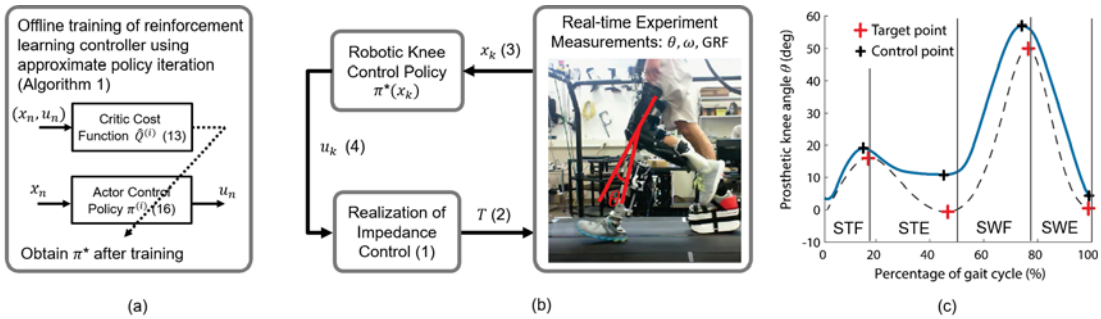

Fig. 1. Overview of offline reinforcement learning controller design and online human subject testing. (a) The offline training process (Algorithm 1). Here x_n and u_n are state and action of the nth offline collected sample, respectively. The optimal policy π^* is obtained after training. (b) The online testing process. State x_k is computed based on real-time measurements, then action u_k, i.e., the adjustment to the impedance parameters, is computed according to the offline trained policy $\pi^*(x_k)$. Finally, according to the well established FSM framework, a knee joint torque T is created based on the impedance control law (2). (c) Target points and control points are defined on gait trajectories. The grey dashed line shows knee kinematics of normal human walking and the blue line represents actual measured knee kinematics. The red crosses are target points in the normal knee kinematics and black crosses are control points of measured knee kinematics. State x_k is formulated using the vertical and horizontal distances between the control points and the target points.

usually cannot last more than one hour because of human fatique and safety considerations. Putting it together, we are in need of a reinforcement learning controller that can adapt to individual conditions in a timely and data efficient manner.

In our previous study [26], [27], we developed an actor critic RL controller, namely direct heuristic dynamic programming (dHDP) [28] to the robotic knee prosthesis parameter tuning problem. By interacting with the human-prosthesis system and under the same FSM impedance control framework, dHDP learned to reproduce near-normal knee kinematics. Though the dHDP showed its promise, it still took a relatively long time to complete the learning process. It took about 300 gait cycles or about 10 minutes of walking to achieve acceptable walking performance [26]. Moreover, because it is an online learning algorithm, it has not been developed to take advantage of existing offline data. Therefore, the problem calls for a more time efficient and data efficient solution.

To this end, we introduce an innovative, approximate policy iteration based reinforcement learning controller. Compared to the previous dHDP approach, it has several advantages. First, it enjoys several important properties of classic policy iteration algorithm such as convergent value functions and stable iterative control policies [29], [30]. Second, it is reported that policy iteration has higher data and time efficiency than general gradient descent based methods [31]. Third, as we aim to show in this paper that our policy iteration based RL approach can learn from offline data to fully utilize historical data. As such, this learning controller can potentially be expanded to solve more complex problems that require an integration of both online and offline data.

The objective of this study is to develop and evaluate the feasibility of a policy iteration based learning control for personalizing a robotic prosthesis. In our previous study [32], we conducted a simulation study to indicate the potential of the proposed idea. Our approach is based on that in [29], which is further developed in this study to provide real time control for a real physical robotic prosthesis with human in the loop. The real human-prosthesis system is rich in unmodeled dynamics and uncertainties from environment and human. Especially, the human variances and consequent impact on the prosthetic knee and the human-prosthesis system have made controlling the robotic prosthesis more challenging than those problems encountered in humanoid robots or human-robot interactions to jointly perform a task such as picking up a box. This is because the human-prosthesis system interact and evolve seamlessly at an almost instantaneous time scale, i.e., a potentially out-of-control parameter adjustment in the prosthesis can result in system instability almost immediately, which is much less tolerant than a human-robot system.

In this paper, for the first time, we successfully designed a reinforcement learning controller realized by approximate policy iteration to control robotic lower limb prosthesis with human in the loop. This new prosthesis control design approach is data efficient as it was derived from offline data collected from interactions between human and prosthesis. We demonstrated this learning controller for tuning 12 prosthesis parameters to approach desired normal gait on real human subject.

II. Human-Prosthesis Integrated System

A. Finite State Machine Framework

Fig. 1 illustrates reinforcement learning controlled prosthesis in a human-prosthesis integrated system. The learning controller is realized within a well established FSM platform. Specifically, an FSM partitions a gait cycle into four sequential gait phases based on knee joint kinematics and ground reaction force (GRF). These four gait phases are stance flexion (STF), stance extension (STE), swing flexion (SWF) and swing extension (SWE). In real-time experiments, transitions between phases are realized as those in [11] based on Dempster-Shafer theory (DST). For each phase, the prosthetic system mimicked a passive spring-damper-system with predefined impedance that matched the biological knee impedance. The predefined impedance parameters are selected by the finite state machine and outputted to the impedance controller as

$$I = [K, B, \theta_e]^T \in \mathbb{R}^3, \tag{1}$$

where K is stiffness, B is damping coefficient and θ_e is equilibrium position. In other words, for all four phases there are 12 impedance parameters to activate the knee joint which directly impact the kinematics of the robotic knee and thus the performance of the human-prosthesis system. The knee joint torque $T \in \mathbb{R}$ is generated based on the impedance control law

$$T = K(\theta - \theta_e) + B\omega. \tag{2}$$

The four target points (red markers) and four control points (black markers) in Fig. 1(c) provide state information for the learning controller to generate optimal control. The chosen points were the maximum or minimum points within each phase, so they could characterize basic knee movements. To approach the normal gait, target points were set to resemble the corresponding points in normative knee kinematics measured in able-bodied individuals [33].

Specifically, one learning controller is designed for one phase under the FSM framework. Without loss of generality, our following discussion involves only one of the four phases. In each phase, peak error $\Delta P \in \mathbb{R}$ and duration error $\Delta D \in \mathbb{R}$ are defined as the vertical and horizontal distance between the corresponding pair of control point and target point. Then the state x of the RL controller are formed using $\Delta P \in \mathbb{R}$ and $\Delta D \in \mathbb{R}$ as

$$x = [\Delta P, \Delta D]^T. \tag{3}$$

Correspondingly, the action u is the impedance adjustment ΔI,

$$u = \Delta I. \tag{4}$$

Additional insights and construct on the FSM framework and the peak/duration errors can be found in [27].

III. Offline Reinforcement Learning Control Design

A. Problem Formulation

In this paper, we consider the integrated human-prosthesis system as a discrete-time nonlinear system (5),

$$x_{k+1} = F(x_k, u_k), k = 0, 1, 2, \dots \tag{5}$$

$$u_k = \pi(x_k) \tag{6}$$

where k is the discrete time index that provides timing for each impedance control parameter update, $x_k \in \mathbb{R}^2$ is the state vector x at time k, $u_k \in \mathbb{R}^3$ is the action vector u at time k, F is the unknown system dynamics, and $\pi : \mathbb{R}^2 \to \mathbb{R}^3$ is the control policy.

To provide learning control of the prosthesis within system (5), we formulate an instantaneous cost function $U(x, u)$ in a quadratic form as

$$U(x, u) = x^T R_x x + u^T R_u u \tag{7}$$

where $R_x \in \mathbb{R}^{2\times 2}$ and $R_u \in \mathbb{R}^{3\times 3}$ are positive definite matrices. We use (7) to regulate state x and action u, as larger peak/duration error as in (3) and larger impedance adjustment as in (4) will be penalized with a larger cost.

The infinite horizon cost function $Q(x_k, u)$ is defined as

$$Q(x_k, u) = U(x_k, u) + \sum_{j=k+1}^{\infty} \gamma^{j-k} U(x_j, \pi(x_j)) \tag{8}$$

where γ is a discount factor. Note that the $Q(x_k, u)$ represents the cost function when action u is applied at state x_k, the system (5) then reaches x_{k+1} and follows the control policy π thereafter.

The optimal cost function $Q^*(x_k, u)$ satisfies the Bellman optimality equation

$$Q^*(x_k, u) = U(x_k, u) + \gamma Q^*(x_{k+1}, \pi^*(x_{k+1})) \tag{9}$$

where the optimal control policy $\pi^*(x_k)$ can be determined from

$$\pi^*(x_k) = \arg\min_u Q^*(x_k, u). \tag{10}$$

Policy iteration is used to solve the Bellman optimality equation (9) iteratively in this study. Policy iteration has several favorable properties such as convergence guarantee and high efficiency [29], which make it a good candidate for configuring a robotic knee with human in the loop. Starting from an initial admissible control $\pi^{(0)}(x_k)$, the policy iteration algorithm evolves from iteration i to $i+1$ according to the following policy evaluation step and policy improvement step. Note that for offline training, a zero output policy is sufficient to be an initial admissible control.

Policy Evaluation

$$Q^{(i)}(x_k, u) = U(x_k, u) + \gamma Q^{(i)}(x_{k+1}, \pi^{(i)}(x_{k+1})) \quad i = 0, 1, 2, \dots \tag{11}$$

Policy Improvement

$$\pi^{(i+1)}(x) = \arg\min_u Q^{(i)}(x, u), i = 0, 1, 2, \dots \tag{12}$$

Equation (11) performs an off-policy policy evaluation, which means the action u need not to follow the policy being evaluated. In other words, $u \neq \pi^{(i)}(x_k)$ in general. This makes it possible to implement (11) and (12) in an offline manner using previously collected samples and thus achieve data efficiency. Solving (11) and (12) requires exact representations of both cost function and control policy, which is often not tractable in robotic knee configuration problem where continuous state and continuous control are involved. In Subsect. III-B, we circumvent this issue by finding an approximated solution for (11) using offline data.

B. Offline Approximate Policy Iteration

For implementation of the policy evaluation equation (11), we used a quadratic function approximator to approximate the cost function $Q^{(i)}(x, u)$ in the ith iteration as

$$\hat{Q}^{(i)}(x, u) = \begin{bmatrix} x \\ u \end{bmatrix}^T S^{(i)} \begin{bmatrix} x \\ u \end{bmatrix} = \begin{bmatrix} x \\ u \end{bmatrix}^T \begin{bmatrix} S^{(i)}_{xx} & S^{(i)}_{xu} \\ S^{(i)}_{ux} & S^{(i)}_{uu} \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} \tag{13}$$

where $S^{(i)} \in \mathbb{R}^{5\times 5}$ is a positive definite matrix and $S^{(i)}_{ux}, S^{(i)}_{xx}, S^{(i)}_{xu}$ and $S^{(i)}_{uu}$ are submatrices of $S^{(i)}$ with proper

TABLE I

BOUNDS ON THE ACTIONS

Gait Phase	K $(N{\cdot}m/deg)$	θ_e (deg)	B $(N{\cdot}m{\cdot}s/deg)$
STF	$[-0.1, 0.1]$	$[-1, 1]$	$[-0.001, 0.001]$
STE	$[-0.1, 0.1]$	$[-1, 1]$	$[-0.001, 0.001]$
SWF	$[-0.01, 0.01]$	$[-2, 2]$	$[-0.001, 0.001]$
SWE	$[-0.01, 0.01]$	$[-1, 1]$	$[-0.001, 0.001]$

dimensions. The quadratic form of (13) corresponds to the instantaneous cost function $U(x, u)$ in (7).

To utilize offline data with the approximated cost function (13), samples are formulated as 3-tuples $(x_n, u_n, x'_n), n = 1, 2, 3 \ldots N$, where n is the sample index and N is the total number of samples of the offline dataset. The 3-tuple (x_n, u_n, x'_n) means that after control action u_n is applied at state x_n, the system reaches the next state x'_n. In other words, $x_n \xrightarrow{u_n} x'_n$ is required to formulate a sample, but x'_n needs not to equal to x_{n+1} and u_n does not need to be on-policy, i.e. following a specific policy. Notice that k represents a sequential time evolution associated with gait cycle, but n does not need to follow such an order because offline sample n and $n + 1$ may come from two different trials. Hence, collecting offline samples is much more flexible than collecting online learning samples. Having an offline dataset $D = \{(x_n, u_n, x'_n), n = 1, 2, 3 \ldots N\}$, we can perform the following approximate policy evaluation step according to (11),

$$\hat{Q}^{(i)}(x_n, u_n) = U(x_n, u_n) + \gamma \hat{Q}^{(i)}(x'_n, \pi^{(i)}(x'_n)). \quad (14)$$

Solving (14) for $\hat{Q}^{(i)}(x_n, u_n)$ is equivalent to solving for $S^{(i)}$. In other words, based on (13), the policy evaluation (14) can be converted to the following convex optimization problem with respect to $S^{(i)}$,

$$\begin{aligned} &\text{minimize } \mu_n^T S^{(i)} \mu_n - \gamma (\mu'_n)^T S^{(i)} \mu'_n - U(\mu_n) \\ &\text{subject to } S^{(i)} \succ 0 \end{aligned} \quad (15)$$

where $\mu_n = [x_n^T, u_n^T]^T$ and $\mu'_n = [x'^T_n, \pi^{(i)}(x'_n)^T]^T$. After obtaining the $S^{(i)}$ and $\hat{Q}^{(i)}(x_n, u_n)$, we can update policy based on

$$\pi^{(i+1)}(x_n) = \arg\min_{u_n} \hat{Q}^{(i)}(x_n, u_n) \quad (16)$$

which is an approximate version of (12). In practice, constraints on actions are added to keep actions within a reasonable range (TABLE I). As a result, policy update (16) can be converted to a quadratic programming problem,

$$\begin{aligned} &\text{minimize } \hat{Q}^{(i)}(x_n, u_n) \\ &\text{subject to } u_- \leqslant u_n \leqslant u_+ \end{aligned} \quad (17)$$

where u_- and u_+ are the lower bound and upper bound of acceptable action, respectively. The values of u_- and u_+ can be found in TABLE I. We used convex optimization [34] to solve (15) and (17).

Algorithm 1 summarizes the implementation of the offline approximate policy iteration algorithm.

Algorithm 1 Offline Approximate Policy Iteration

Input: training dataset $D = \{(x_n, u_n, x'_n), n = 1, 2, \ldots, N\}$
Output: optimal cost function $\hat{Q}^*(x, u)$ and policy $\pi^*(x_k)$

1: **for** $i = 1, 2, \ldots, i_{max}$ **do**
2: Get $S^{(i)}$ from (15) and $\hat{Q}^{(i)}(x, u)$ from (13)
3: Get policy $\pi^{(i+1)}(x)$ from (17)
4: **end for**
5: **return** $\hat{Q}^*(x, u) = \hat{Q}^{(i)}(x, u)$ and $\pi^*(x) = \pi^{(i+1)}(x)$

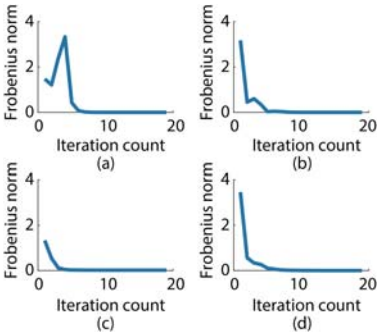

Fig. 2. The Frobenius norm of the difference between two successive S matrices which vary as the policy iteration number increases for the four different phases. (a) Stance flexion. (b) Stance extension. (c) Swing flexion. (d) Swing extension.

C. Implementation of Offline Policy Training

The offline training data including $N = 140$ pairs of the (x_n, u_n, x'_n) tuples came from two separate experiments invovling the same human subject using the same prosthesis device. The whole data collection process took 29 minutes to complete. During data collection, the prosthesis impedance parameters were controlled by the dHDP based RL approach that we investigated previously [26]. Note, however, that the dHDP was used to only provide some control to the prosthesis or in other words, dHDP was an enabler of the data collection session. That is to say that the data were drawn from the online learning process of the dHDP RL controller rather than generated by a well-learned policy. During data collection, the state x_n and next state x'_n in each pair of sampled tuples were averaged by 7 gait cycles conditioned on the same action u_n. In addition, prior to applying Algorithm 1, all samples were normalized into the range between -1 and 1 to avoid ill-conditioning issues during application of convex optimization to achieve admissible control policies.

The discount factor γ was set to 0.8. The termination condition of the Algorithm 1 was set as a maximum of $i_{max} = 100$ iterations. The weight matrices of state and action were specified as $R_x = \text{diag}(10, 1)$ and $R_u = \text{diag}(1, 1, 1)$, respectively. They were specified to make the peak error dominating the cost. Because, compared to the duration error which is partially controlled by human behaviors (e.g. heel-strike or toe-off timing), the peak error is more sensitive to the parameter changes. Moreover, as a factor determining gait performance, the peak error is more important than the action taken in our settings. Yet, we still need to take the duration error as one of the monitored states in the controller, because the controller has to adjust parameters to keep the duration error in a reasonable range. Otherwise, human users

cannot stabilize the duration error by themselves.

To evaluate the convergence of the trained policies, we investigated the changes of S matrix in the approximate cost function $\hat{Q}$ over the entire offline training process for each phase. As a measure of element-wise distance regarding two matrices, the Frobenius norm of the difference between two successive matrices $\left\|S^{(i+1)} - S^{(i)}\right\|_F$ was adopted to quantify the changes. As Fig. 2 shows, the norm value of the difference reduced fast when the training process started off for each phase, and they all approached zeros within 10 iterations. The result indicates that the approximated cost function as well as the policy was convergent and optimal given the training dataset. It took about 5 minutes to perform the offline training until reached the convergence.

IV. Online Human Subject Testing Experiments

A. Experimental Protocol and Setup

The offline trained policy was implemented on the online able-bodied subject testing experiments. The male subject was the same one from whom we collected the offline training data. He was involved with informed consent. The experimental protocol was approved by the Institutional Review Board (IRB) of University of North Carolina at Chapel Hill. During the experiment, the subject wore a powered knee prosthesis and walked on a split-belt treadmill at a fixed speed of 0.6 m/s without holding handrails.

The entire experiment consisted of three sessions with different sets of initial impedance parameters for the prosthetic knee. The three sets of parameters were randomly selected, yet initially feasible to carry on policy iteration. The subject experienced 40 updates of the impedance control parameters for each phase of the FSM during a single experiment session. To reduce the influence of noises introduced by human variance during walking, the update period (i.e., the time index k in (5)) was set as 4 gait cycles (i.e., the states were obtained as an average of every 4 gait cycles). The proposed offline policy iteration based RL controller was used to automatically update impedance control parameters online such that actual knee kinematics approached predefined target points. At the beginning and at the end of each session, the subject had two stages of acclimation walking corresponding to the initial and final set of parameters, respectively. Each stage consisted of 20 gait cycles. The measured knee kinematics in the corresponding acclimation were averaged out to contrast the before-after effects of the proposed controller.

The robotic knee prosthesis used in this study was described in [11]. This prosthesis used a slider-crank mechanism, where the knee motion was driven by the rotation of the moment arm powered by the DC motor through the ball screw. The prosthetic knee kinematics were recorded by a potentiometer embedded in the prosthesis. Some major gait events determining phase transitions in the finite state machine were detected by a load cell. The control system of the robotic knee prosthesis was implemented by LabVIEW and MATLAB in a desktop PC.

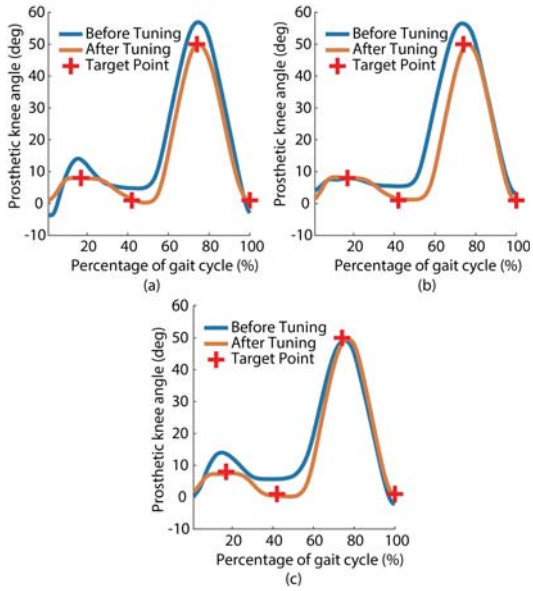

Fig. 3. Three comparisons (corresponding to three different sets of initial impedance parameters) of knee kinematics for before and after impedance parameter tuning. (a) The first set of initial parameters. (b) The second set of initial parameters. (c) The third set of initial parameters.

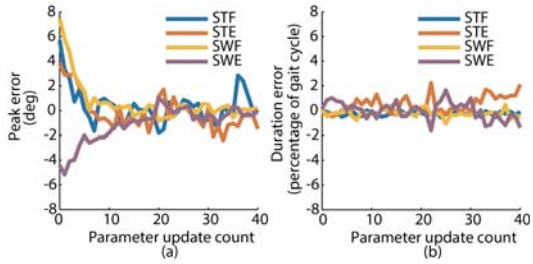

Fig. 4. Evolution of states ((a) peak error and (b) duration error) as impedance parameters were updated. This result corresponds to the case with the first set of initial parameters (i.e., the same initial condition as in Fig. 3(a)).

B. Performance Evaluations

Measures of knee kinematics were obtained at the beginning acclimation stage and at the ending acclimation stage during each session. These measurements reflect how the prosthetic knee joint moved when it interacted with the human subject before and after experiencing the control parameter update. By comparing the respective errors with respect to target points, the performance of the RL controller in a human-prosthesis system can be assessed.

While knee kinematic measures provide a quantitative evaluation of controller performance in terms of reaching desired gait target points, it is also necessary to consider an acceptable error range for the kinematic states. This is because the inherent human variance during walking. Our experiments indicate that when the peak errors and duration errors are within 2 degrees and 2 percent range of the target values, respectively, the human subject would not feel any discomfort or insecure while walking. Therefore, in our study, we adopted those error bounds.

C. Experimental Results

As Fig. 3 shows, the knee kinematics of the initial acclimation stages were different in three different sessions

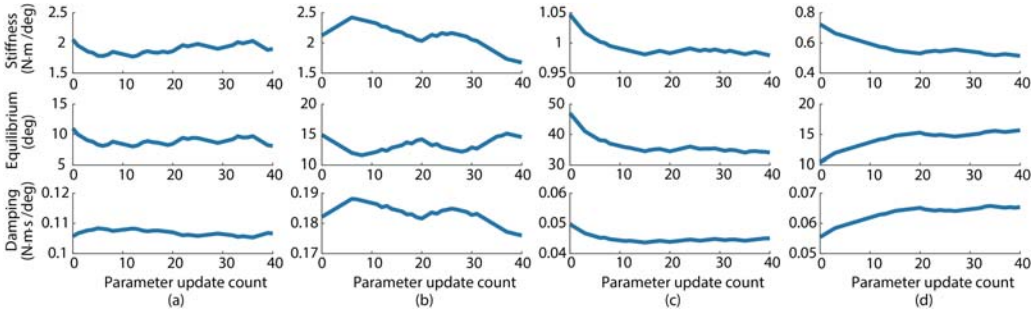

Fig. 5. Evolution of the impedance parameters in different phases (a) STF (b) STE (c) SWF (d) SWE.

and distant from the target points, especially the peak angle errors. Clearly, after the impedance parameters were adjusted by the proposed RL controller, knee kinematics of the final acclimation stages approached the target points. Specifically, the averaged absolute values of the peak errors over the three sessions deceased from 4.18 ± 3.28 degrees to 0.56 ± 0.47 degrees for STF, from 4.33 ± 0.44 degrees to 1.11 ± 0.66 degrees for STE, from 4.92 ± 3.78 degrees to 0.14 ± 0.04 degrees for SWF and from 3.21 ± 1.23 degrees to 0.25 ± 0.23 degrees for SWE. The results indicate that offline policy iteration based RL controller is able to reshape the prosthetic knee kinematics to meet the target points from different initial parameter settings.

Fig. 4 illustrates the evolution of peak errors and duration errors during the experimental session under the first set of initial parameters corresponding to the first result in Fig. 3. Since similar results were obtained from other experiment sessions, hereafter we only present the result from the first session as an example. All four phases experienced reduction in the peak angles errors at the end . Specifically, the peak error decreased from 5.8 degrees to -0.2 degrees for STF, from 3.8 degrees to -1.5 degrees in the STE phase. For SWF and SWE, they dropped from 7.4 degrees to 0.18 degrees and from -4.4 degrees to 0.05 degrees respectively.

The duration errors were insignificant, i.e., they were within the range of two percent of one gait cycle, and they remained within the range over the entire session. There are two considerations in this study. First, the duration time is controlled partially by human behavior, or in other words, the effect of controller on this state at the prosthetic knee is not the exclusively decisive factor. Second, given the previous consideration, we placed more emphasis on the peak error than the duration error as reflected in the weighting matrix R_x in the quadratic cost measure.

The state errors at the final stage are mostly within the bounds of 2 degrees and 2 percent, respectively. These errors remained within bounds thereafter the first 10 parameter update cycles (40 gait cycles, about 1.3 minutes). Compared to the state errors achieved by dHDP [26], the offline policy iteration based RL controller achieved comparable performance with small errors (i.e. ± 2 degrees, ± 2 percent), but with less time to adjust the impedance control parameters. Specifically, it took dHDP 10 minutes of experiment (300 gait cycles) to achieve comparable state errors.

Note that the peak errors from the STF and the STE phases are usually associated with more oscillations than the other two swing phases as the state errors approach zeros (from the 10^{th} update to the 40^{th} update). In addition, as illustrated in Fig. 5, the impedance parameters exhibited different change patterns during the experimental sessions. It is apparent that the impedance parameters during swing phases converged in the first 20 updates and remained stationary thereafter. However, the impedance parameters exhibited somewhat oscillatory patterns during the stance phases. It is actually not surprising when we see the different patterns in the above. As can be understood, the stance phases involve direct interactions and thus directly affected by the ground, the human subject and the robotic prosthesis (for example, loading induced variation). Such varying interactions would introduce more perturbations to the prosthesis and result in oscillations. Whereas the swing phases are less likely to be affected by these factors and thus the state errors during these phases appear more stationary. Under the above discussed disturbances, the RL controller responded by making adjustments when it observed discrepancies between target and actual states. This unique phenomena is a result of us dealing with an inherently co-adapting human-prosthesis system.

V. Conclusion and Future Work

We developed a new data efficient and time efficient approximate policy iteration RL controller to optimally configure impedance parameters automatically for robotic knee prosthesis. The learning controller was trained offline using historical data and then the learned control policy was applied for online control of the prosthetic knee. Our experimental results validated this new approach and showed that it reproduced near-normal knee kinematics for the robotic knee prosthesis. Our results proved that the offline policy iteration based RL controller is a promising new tool to solve the challenging parameter tuning problems for the robotic knee prosthesis with human in the loop.

In this paper, we only collected one subject's data to train the offline policy and tested it on the same subject. Further studies need to be done to investigate whether the outcome of the proposed method can be generalized or transferred to other subjects. In addition, our future work will extend the current design to facilitate further online control policy adjustment. We believe such an integrated approach will facilitate even broader range of human-prosthesis integrated behavior to address changes in environment, task, and human condition.

REFERENCES

[1] S. K. Au, J. Weber, and H. Herr, "Powered ankle-foot prosthesis improves walking metabolic economy," *IEEE Trans. Robot.*, vol. 25, no. 1, pp. 51–66, Jan 2009.

[2] P. Malcolm, R. E. Quesada, J. M. Caputo, and S. H. Collins, "The influence of push-off timing in a robotic ankle-foot prosthesis on the energetics and mechanics of walking," *J. Neuroeng. Rehabil.*, vol. 12, no. 21, Feb 2015.

[3] S. Au, M. Berniker, and H. Herr, "Powered ankle-foot prosthesis to assist level-ground and stair-descent gaits," *Neural Netw.*, vol. 21, no. 4, pp. 654–666, Apr 2008.

[4] A. H. Shultz and M. Goldfarb, "A Unified Controller for Walking on Even and Uneven Terrain with a Powered Ankle Prosthesis," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 26, no. 4, pp. 788–797, Feb 2018.

[5] B. E. Lawson, H. A. Varol, and M. Goldfarb, "Standing stability enhancement with an intelligent powered transfemoral prosthesis," *IEEE Trans. Biomed. Eng.*, vol. 58, no. 9, pp. 2617–2624, Jun 2011.

[6] D. Quintero, D. J. Villarreal, D. J. Lambert, S. Kapp, and R. D. Gregg, "Continuous-Phase Control of a Powered Knee-Ankle Prosthesis: Amputee Experiments Across Speeds and Inclines," *IEEE Trans. Robot.*, vol. 34, no. 3, pp. 686–701, Feb 2018.

[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] R. D. Gregg, T. Lenzi, L. J. Hargrove, and J. W. Sensinger, "Virtual constraint control of a powered prosthetic leg: From simulation to experiments with transfemoral amputees," *IEEE Trans. Robot.*, vol. 30, no. 6, pp. 1455–1471, Oct 2014.

[9] F. Sup, A. Bohara, and M. Goldfarb, "Design and control of a powered transfemoral prosthesis," *Int. J. Rob. Res.*, vol. 27, no. 2, pp. 263–273, Feb 2008.

[10] B. E. Lawson, H. A. Varol, A. Huff, E. Erdemir, and M. Goldfarb, "Control of stair ascent and descent with a powered transfemoral prosthesis," *IEEE Trans. Neural Syst. Rehabil. Eng.*, vol. 21, no. 3, pp. 466–473, Oct 2013.

[11] M. Liu, F. Zhang, P. Datseris, and H. Huang, "Improving Finite State Impedance Control of Active-Transfemoral Prosthesis Using Dempster-Shafer Based State Transition Rules," *J. Intell. Robot. Syst. Theory Appl.*, vol. 76, no. 3-4, pp. 461–474, Dec 2014.

[12] 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, pp. 1611–1625, Oct 2014.

[13] 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.

[14] M. L. Handford, M. Srinivasan, M. Hulliger, and R. Zernicke, "Robotic lower limb prosthesis design through simultaneous computer optimizations of human and prosthesis costs," *Sci. Rep.*, pp. 1–7, Feb 2016.

[15] M. R. Tucker, C. Shirota, O. Lambercy, J. S. Sulzer, and R. Gassert, "Design and Characterization of an Exoskeleton for Perturbing the Knee during Gait," *IEEE Trans. Biomed. Eng.*, vol. 64, no. 10, pp. 2331–2343, Oct 2017.

[16] 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, May 2016.

[17] Y. Ding, M. Kim, S. Kuindersma, and C. J. Walsh, "Human-in-the-loop optimization of hip assistance with a soft exosuit during walking," *Sci. Rob.*, vol. 3, no. 15, Feb 2018.

[18] N. Thatte, H. Duan, and H. Geyer., "A method for online optimization of lower limb assistive devices with high dimensional parameter spaces," in *IEEE Int. Conf. Robot. Autom.*, 2018, pp. 5380 – 5385.

[19] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-End Training of Deep Visuomotor Policies," *J. Mach. Learn. Res.*, vol. 17, no. 1, pp. 1334–1373, Jan 2016.

[20] J. Peters and S. Schaal, "Reinforcement learning of motor skills with policy gradients," *Neural Netw.*, vol. 21, no. 4, pp. 682–697, May 2008.

[21] S. Levine, N. Wagener, and P. Abbeel, "Learning contact-rich manipulation skills with guided policy search," in *IEEE Int. Conf. Robot. Autom.*, 2015, pp. 156–163.

[22] I. Mordatch, N. Mishra, C. Eppner, and P. Abbeel, "Combining model-based policy search with online model learning for control of physical humanoids," in *IEEE Int. Conf. Robot. Autom.*, Stockholm, 2016, pp. 242–248.

[23] S. Gu, E. Holly, T. Lillicrap, and S. Levine, "Deep reinforcement learning for robotic manipulation with asynchronous off-policy updates," in *Proc. - IEEE Int. Conf. Robot. Autom.*, 2017.

[24] W. Montgomery, A. Ajay, C. Finn, P. Abbeel, and S. Levine, "Reset-free guided policy search: Efficient deep reinforcement learning with stochastic initial states," in *IEEE Int. Conf. Robot. Autom.*, 2017, pp. 3373–3380.

[25] D. Vogt, S. Stepputtis, S. Grehl, B. Jung, and H. Ben Amor, "A system for learning continuous human-robot interactions from human-human demonstrations," in *Proc. - IEEE Int. Conf. Robot. Autom.*, 2017.

[26] Y. Wen, J. Si, A. Brandt, X. Gao, and H. Huang, "Online reinforcement learning control for the personalization of a robotic knee prosthesis," *IEEE Trans. Cybern.*, Jan 2019, doi:10.1109/TCYB.2019.2890974.

[27] Y. Wen, J. Si, X. Gao, S. Huang, H. Huang, and S. Member, "A New Powered Lower Limb Prosthesis Control," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 28, no. 9, pp. 2215 – 2220, Jul 2017.

[28] J. Si and Y. T. Wang, "On-line learning control by association and reinforcement," *IEEE Trans. Neural Networks*, vol. 12, no. 2, pp. 264–276, Mar 2001.

[29] W. Guo, F. Liu, J. Si, D. He, R. Harley, and S. Mei, "Online Supplementary ADP Learning Controller Design and Application to Power System Frequency Control With Large-Scale Wind Energy Integration," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 27, no. 8, pp. 1748–1761, Jun 2016.

[30] W. Guo, J. Si, F. Liu, and S. Mei, "Policy Approximation in Policy Iteration Approximate Dynamic Programming for Discrete-Time Nonlinear Systems," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 29, no. 7, pp. 2794 – 2807, Jul 2018.

[31] M. G. Lagoudakis and R. Parr, "Least-squares policy iteration," *J. Mach. Learn. Res.*, vol. 4, pp. 1107–1149, Dec 2003.

[32] X. Gao, J. Si, Y. Wen, M. Li, and H. He, "Adaptive Batch Policy Iteration Parameter Optimization for Powered Prosthetic Knee with Human in the Loop," *Submitt. to IEEE Trans. Neural Networks Learn. Syst.*, 2018.

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

[34] M. Grant and S. Boyd, "CVX: Matlab software for disciplined convex programming, version 2.1," http://cvxr.com/cvx, Mar. 2014.