



# Hierarchical Human Machine Interaction Learning for a Lower Extremity Augmentation Device

Likun Wang<sup>1,2</sup> · Zhijiang Du<sup>1</sup> · Wei Dong<sup>1</sup> · Yi Shen<sup>2</sup> · Guangyu Zhao<sup>3</sup>

Accepted: 28 May 2018 / Published online: 4 June 2018  
© Springer Science+Business Media B.V., part of Springer Nature 2018

## Abstract

For several years considerable effort has been devoted to the study of human augmentation robots. Traditionally, the focus of exoskeleton system has always been on model-based control framework. It seeks to model the dynamic system from prior knowledge of the robot as well as the pilot. However, in lower extremity exoskeleton, the control method depends on not only the modelling accuracy but also the physical human–machine interaction changed from personal physical conditions. To address this problem, in this paper, we present a model-free incremental human–machine interaction learning methodology. In a higher level, the methodology can plan the motion of exoskeleton with the sequence of rhythmic movement primitives. In the lower level, the gain scheming is updated from the dynamic system based on a novel proposed learning algorithm efficient  $PI^2$ -CMA-ES. Compared with  $PI^{BB}$ , a particular feature is that it directly operates on the Cholesky decomposition of the covariance matrix, reducing the computational effort from  $O(n^3)$  to  $O(n^2)$ . To evaluate our proposed methodology, we not only demonstrate its applications on the single leg exoskeleton platform but also test on our lower extremity augmentation device. Experimental results show that the proposed methodology can minimize the interaction between the pilot and the exoskeleton compared with the traditional model-based control strategy.

**Keywords** Exoskeleton · Rhythmic movement primitives (RMPs) · Reinforcement learning ·  $PI^2$  · CMA-ES · Human machine interaction (HMI)

## 1 Introduction

Quite recently, considerable attention has been paid to the powered exoskeleton, both for the lower and upper extrem-

ities [1–5]. In the related application, the lower extremity exoskeleton has been shown great potential, ranging from human strength augmentation to medical assistance as well as motor rehabilitation. In all those cases, the principal characteristics are designed to mimic human movement primitives. To follow the human intention precisely, the exoskeleton should be able to detect human motion intention with little interaction between the exoskeleton and the pilot. In order to address this crucial issue, current research on motion capture is focused on two types controllers, i.e. model-based controller and sensor-based controller.

The core idea of sensor-based controllers is that the input to the controller is collected from the sensor system. Besides, by applying many variations of control strategies, such as impedance control and master-slave control [6], the robotic system can follow the real-time human intention. For instance, with impedance control, the Hybrid Assistive Limb (HAL) exoskeleton is driven by Electro-Myo-Graphical (EMG) sensors data [7]. In their experiments, the pilot's intention is recognized from different reference patterns through measurement from the EMG sensors directly. Addi-

✉ Wei Dong  
dongwei@hit.edu.cn

Likun Wang  
likunwang@hit.edu.cn

Zhijiang Du  
duzj01@hit.edu.cn

Yi Shen  
shen@hit.edu.cn

Guangyu Zhao  
guangyuzhao1980@gmail.com

<sup>1</sup> State Key Laboratory of Robotics and System, Harbin Institute of Technology, Harbin 150080, People's Republic of China

<sup>2</sup> School of Astronautics, Harbin Institute of Technology, Harbin 150080, People's Republic of China

<sup>3</sup> Weapon Equipment Research Institute, China Ordnance Industries Group, Beijing 102202, People's Republic of China

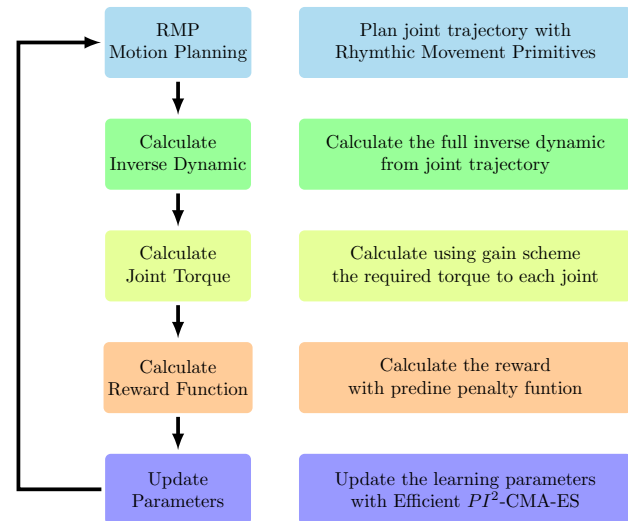
tionally, fuzzy impedance control [8] and active impedance control [9] are designed for adaptation to the changing of the interaction dynamics, measured by the force sensor mounted between the human and the exoskeleton. The main advantage of this type of controllers is that it does not depend on the model, which may bring some benefits for control system design. However, the obvious limit is that the robotic control quality relies heavily on the complicated sensors system.

An alternative is based on the human–robot coupling dynamic model. One of this kind of solution is proposed in [10], and the control method is so-called Sensitivity Amplification Control (SAC) which is applied in Berkeley Lower Extremity Exoskeleton (BLEEX) [11]. As for the control scheme, the sensitivity factor is utilized to make the system more sensitive to the interaction. Thus the interaction force is naturally minimized. Without the need of the sensors to directly measuring the interaction force, the SAC control strategy simplifies the sensors system. Although the complexity of the sensor system is reduced, this methodology has a strong dependence on the accuracy of the dynamic models. Since the dynamic model is quite complicated, the identification process for the model parameters is never an easy task [12]. Furthermore, in [13], Mitrovic et al. analysis a strong case for the limitations of dynamic models. However, SAC has high sensitivity to the human motion, but also to the environment. Therefore if the undesired disturbance acts on the device, the exoskeleton will response the undesired motion intention.

Most of the control methodology mentioned above is characterized by the negative error feedback control with high gain, just like the traditional robotics control strategy. However, when pilot interacts with an exoskeleton, a high gain may lead to unsafety and energy consuming. Thus, the concept of adaptive impedance with the time-varying derivative and proportional gains can be considered as an alternative. However, “the selection of good impedance parameters [...] is not an easy task” [14].

To address this issue, one possible solution is the optimal control [15,16], where its gain scheduling is automatically selected via many optimal algorithms. However, it still relies on the model-based derivations. Thus, in the complex environment where the prior model’s knowledge is not clear enough, such algorithms are not applicable. Closely related to optimal control is Reinforcement Learning (RL) [17]. The main idea of RL is that, through error and trial, the learning algorithm updates the parameters with minimizing reward function. As the main advantage, the algorithm explores the environment without prior knowledge. Nevertheless, RL suffers high-dimensional curse in continuous state control.

From the point view of biological movements, we hope that the exoskeleton acts as human, in another, the exoskeleton should ensure the safe interaction with a human, which implies that the impedance control strategy should be



**Fig. 1** The proposed hierarchical human–machine interaction learning methodology for a lower augmentation device control

time-varying and compliant. Consequently, we consider the exoskeleton control as a biological motor system, which has some advantages over the other control strategies regarding robustness, performance, and versatility. Thus, for exploring the possibility of inheriting the merits of sensor-based and model-based controller together, a novel Human Machine Incremental Learning strategy is proposed, which is introduced in Fig. 1. A novel designed exoskeleton so-called lower extremity augmentation device (LEAD) is applied to address these crucial issues in this paper, as shown in Fig. 2. The strategy is free of dynamic model and does not depend on the complicated sensor system. On the other hand, the methodology seeks to implement a low time-varying impedance gain scheduling.

Not akin to the excellent work in [18,19], the essential difference is that we aim to learn the human motion primitives in real-time with rhythmic movement primitives, which is more natural since the normal human locomotion is also periodic. Moreover, the Q reinforcement learning algorithm is easily suffered from the curse of dimensionality and mainly for the discrete system. In this paper, we aim to provide an online learning algorithm based on the Monte Carlo sampling methods and address the issues in the continuous region of interest.

The *main contributions* of this paper are given as follows:

1. The hierarchical learning framework consists of high-level motion planning and low-level impedance controller is introduced in this paper.
2. We propose a human locomotion model with RMPs as well as sparse pseudo-input Gaussian process regression.
3. A novel model-free reinforcement learning methodology, so-called Efficient  $PI^2$  CMA-ES, is presented in this



**Fig. 2** Lower extremity augmentation device (LEAD) and pilot. From the point view of biomechanical consideration, exoskeleton can be seen as an anthropomorphic device which is parallel with the human body. In the design of exoskeleton, the comfortable locomotion is achieved by designing the number of DOF to be close to humans. Thus, our exoskeleton is designed as follows: the knee joint is actuated by a DC motorized ball screw; the hip joint in the sagittal plane is also an active joint, which is driven by a disk-type motor; the other joints are passive with elastic components due to the simplification of the control system and the consideration of total mass

paper for the purpose of learning the motion trajectory online.

4. Finally, we demonstrate that with the proposed learning algorithm, the compliance between the exoskeleton and human can be adapted and simultaneously, the desired trajectory can be followed.

The remainder of this paper is organized into five sections, after the introduction, Sect. 2 outlines related works of Human–Machine Interaction and reinforcement learning based on stochastic optimal control. Section 3 is devoted to discussing the human–machine coupling model as well as Rhythmic Movement Primitives (RMPs). The main routine of the novel reinforcement learning algorithm, so-called Efficient  $PI^2$ -CMA-ES is presented in Sect. 4. Section 5 compares the proposed learning methodology with several state-of-the-art algorithms and summarizes the results of our simulation work.

## 2 Related Works and Background

### 2.1 Human Machine Interaction and Impedance Control

With the aim to maintain safe physical interaction and achieve robustness towards disturbance, exoskeleton should adapt impedance with biomechanical system [20] and improve user’s motion agility [9].

In tradition robotics system, the control methodology is often with constant high gain parameters, as well as negative feedback control loop. Thus, to achieve high accuracy trajectory tracking leads to massive energy consumption. Especially for industrial robots, the safety of staffs is guaranteed with cages built near the robots. As for autonomous mobile robots, energy consumption is a standard criterion that quantitatively evaluates the ability of robots.

We consider exoskeleton as an autonomous mobile legged robot that is driven by human intention or human movement primitives. The safety of the pilot is the first factor that be carefully considered by the designer. Furthermore, most of the exoskeletons are required to work in the environment with limited resource. Especially for military applications, the pilot is asked for long hiking and executing the task in a rugged environment.

A distinctive characteristic is that the interaction is not only defined as an input to the controller, but also a factor that we want to get close to its minimal. A direct and proper choice dealing with it is applying impedance control methodology [21,22], which maps the force according to the difference between desired state and real state. A typical formulation with stiffness  $K$ , damping  $D$ , mass  $M$  as follows

$$F = M(\ddot{x} - \ddot{x}_d) + D(\dot{x} - \dot{x}_d) + K(x - x_d). \quad (1)$$

For two decades, impedance control has been combined with other control methods especially in feedback loop [23,24]. However establishing a perfect dynamic model is not an easy task, since mechanical friction, system disturbance, and signal noise exist all the time. Moreover, deriving a good impedance controller requires knowledge of both the environment, robot and also well understanding of the system parameters.

### 2.2 Reinforcement Learning in High Dimensions

In recent years, reinforcement learning (RL) has shown great potential in autonomy, adaptivity as well as the flexibility of robots regarding specific tasks. Furthermore, RL can be derived from various aspects, such as stochastic optimal control, dynamic programming, probability exploration and policy gradient improvement. With the purpose of generating more scalable algorithms with fewer variable and higher effi-

ciency, RL trends towards combining traditional techniques from dynamic programming and optimal control with current learning algorithms from statistic estimation.

In terms of finite horizon optimal control problems, Differential Dynamic Programming (DDP) [25] is one of the most popular algorithms, which combines model-based RL and optimal control. In DDP, detectability and stabilizability are of great importance for local dynamics approximation concerning convergence. The control policy consists of closed-loop gains and open feedforward loop parameters. Nevertheless, the space state trajectory is optimized locally. As a result, the DDP cannot cope both with planning and gain scheduling. A computation improvement is suggested in [26]. However, this algorithm can only be used in low-dimensional problems.

In [27], the author proposes a min-max Differential Game Theory approach with robotics application. In essence, the approach is a combination of  $H_\infty$  control and Differential Game Theory. Although the feedback control is robust to model and dynamics uncertainty, it might result in over conservative control policies. As for the linear system, the robustness of the algorithm is feasible with  $\beta$ -iteration, while for the nonlinear system, the robustness is not guaranteed.

An alternative algorithm so-called Receding Horizon DDP is provided in [28] with rather an efficient way of solving the local optimal control problem. However, the optimal trajectories and control computing is off-line. The work on LOR-trees is a variation of DDP, which is based on state space approximation. The model-based approach, so-called iterative Linear Quadratic Regulator (iLQR) [29] can improve sampling using control funnels. Despite the improvement in sampling, a key limitation is that the high dimensional dynamical problem is not addressed.

The work by Todorov [30–32] on stochastic optimal control presents that the Bellman equation can be defined as the Kullback-Leibler divergence for the discrete optimal control problems. The most interesting aspects are that this kind of problems is equivalent to dealing with continuous state dynamics with quadratic value function and under Gaussian noise. In [30], the stochastic optimal control problem is investigated in discrete state dynamics, and thus it is defined as a Markov Decision Process.

The learning methodology proposed in this paper so-called Efficient  $PI^2$ -CMA-ES, is based on  $PI^2$  algorithm and CMA-ES framework. The  $PI^2$  can explore in continuous state space and perform policy improvement concerning the quality of the solution and convergence speed, such as REINFORCE [33] and Natural Actor-Critics [34]. The advantage of  $PI^2$  over the others is that not like the DDP, it does not suffer from the curse of dimensions and is totally model-free. An improvement of  $PI^2$  is introduced in [35], so-called  $PI^{BB}$ . The original  $PI^2$  is constrained with the  $\sigma = \lambda R^{-1}$ , where  $R$  is the cost matrix, and  $\lambda$  is a constant proportional to  $1/h$ .

Thus, based on the Monte Carlo roll-outs, the exploration noise is sampled from a constant covariance matrix. However, the  $PI^{BB}$  is black-box optimization algorithm, which can be seen as a ‘‘Covariance Matrix Adaptation - Evolutionary Strategy’’.

### 2.3 Dynamical Movement Primitives with Central Pattern Generators

Modelling a nonlinear system is rather complicated due to complex state transitions in response to parameter updates, difficulty in coping with prediction and analysis of long-term behaviour. Thus from the traditional view, coming up with a robust control solution for agility exoskeleton system [11,36] is not an easy task, since the human biological motor system and human-machine interaction add a strong nonlinearity to our control system.

For periodic human locomotion, neural and phase oscillators are widely applied regarding a set of Degrees of Freedom (DOF). Central Pattern Generators (CPGs) are typical effective alternative solution to rhythmic movement [37,38]. The main advantage of CPGs is that it is not necessarily taking the system’s dynamics and control dimensionality into account. Besides the robustness of the system and smooth rhythmic motion patterns can be easily gained from sensor data [39,40].

However, the necessary proper parametrization of CPGs is difficult, since the state-space designing process is strenuous. Also note that, although by applying biomechanics and reduction of the active DOFs, the dimensionality can be simplified. Nevertheless, the prior of tuning process knowledge is still required.

The Rhythmic Movement Primitives (RMPs) are nonlinear dynamical systems that generate rhythmic movements by limit cycle attractor as well as means of points [41]. By using attractor dynamics, RMPs are able to design complicated parameterized trajectories which are robust against external disturbance and easily modulated. Thus, it has been used for trajectories generating, obstacle avoiding. However, when dealing with high-dimension state space, the learning process of RMPs weights is still complex [42].

To address this issue, we seek to combine RMP and CPG together, in which we use RMP’s design rule to hand-tuned CPGs. Thus, a canonical system is employed to generate asynchronous convergence state space. Moreover, a transformation system integrates the reference trajectories, and a function approximator forms a smooth desired trajectory. Furthermore, the force term of the approximator modifies the attractor landscape of control strategy and forms a CPG trajectory. To sum up, the proposed solution is independent of initial condition and adapts to new situation which can be designed online.



### 3 Locomotion of Exoskeleton

In this section, the dynamic learning model is established based on variable impedance control and RMPs, which is an extended work of [43]. Firstly, we focus on the HMI modelling that obeys rigid body dynamics. Secondly, A sequence movement primitives with human–exoskeleton locomotion are planned by phase resetting and frequency adaptation.

#### 3.1 Dynamical Model Representation

Without loss of generality, the exoskeleton dynamic is represented in a general formulation given as

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + P(q) = \tau, \tag{2}$$

where  $M(q)$  is defined as a symmetric inertial matrix;  $C(q, \dot{q})$  consists of the Centrifugal and Coriolis matrix;  $P(q)$  is the gravitational matrix and  $\tau$  is defined as the actuated torque.

As there exists difficulty in dealing with exoskeleton dynamics in the above formula, we transform the equations into state space

$$\dot{q}^v = M(q_p)^{-1}(-C(q_p, q_v) - P(q_p) + \tau) \tag{3}$$

$$\dot{q}^p = q^v, \tag{4}$$

The torque term can be written as follows

$$\tau_i = K_{P,i}(q_{d,i}^p - q_i^p) + K_{D,i}(q_{d,i}^v - q_i^v) + u_{com,i}, \tag{5}$$

where  $K_{P,i}$  and  $K_{D,i}$  are positive-definite variable, which represents as stiffness variable and damping variable, respectively. With the computed torque method, a feedforward term  $u_{com,i}$  is added to the control input in order to eliminate the nonlinearity in the dynamics. Based on Lagrange algorithm,  $u_{com,i}$  only compensates the forces due to Coriolis acceleration, gravity and inertia terms, but not the interaction force between pilot and exoskeleton. Thus the learning interaction force is independent of  $u_{com,i}$ .

Based on the principle of impedance control [21], in [43] the connection between the damping and stiffness parameters is defined as  $K_{D,i} = \xi\sqrt{K_{P,i}}$ , where  $\xi$  is a positive constant.

Note that, although our proposed learning methodology is model-free, the system controller does require a model of an exoskeleton. It is well explained in [44], “If the movement would involve both the compensation of the task dynamics and arm dynamics this generalization would not be possible”. Also, note that the human reduces the effect of dynamics of their arm by using feed-forward control [45] since the exoskeleton’s performance should not depend on the specific human subject with different inverse dynamic model.

In brief, the impedance is parameterized by the gain scheme, i.e. stiffness  $K_P$  and damping  $K_D$ . The main idea of interaction control is to define  $K_P$  as a function of implemented movement.

#### 3.2 Interaction Model Representation

Since RMPs are employed to follow human joint trajectories, the exoskeleton device should provide the necessary information about human movement primitives. However, as suggested by [11], the movement intention of the pilot should not be acquired from sensors mounted on the skin such as sEMG sensors. That kind of sensors has a strong dependence on the location of the skin. Moreover, it is not convenient for the pilot to wear the sEMGs sensors whenever he or she wants to use the exoskeleton device. Therefore, we propose a nonlinear regression model to provide the required information for exoskeleton locomotion by applying sparse Pseudo-input gaussian processes (SPGPs) [46]. This model can be interpreted as a mapping from the interaction force to the joint angles in real time.

In our setting, when learning the regression model, the input of  $N$  training sets is defined as  $X = \{q_{k-1}, F\}$ , with  $q_{k-1}$  and  $F$  denoting joints angles and interaction force respectively at time  $k - 1$ . While the corresponding output is given as  $y = \{q_k\}$ . Assuming the mapping function  $f(x)$  corrupted by Gaussian noise  $\mathcal{N}(0, \beta^{-1})$ , the generative model is given below

$$f(X) \sim GP(0, k(X, X)), \quad y \sim \mathcal{N}(f, \beta^{-1}I). \tag{6}$$

Since the function  $f$  should be marginalized out to find the marginal likelihood and predictive distribution, the computation expense related to an inversion  $n \times n$  matrix requires  $O(n^3)$  time complexity.

To address this issue, we employ a computationally tractable regression method, so-called SPGPs mentioned above. The SPGPs aims to construct the regression function with a set of  $m$  ( $m < N$ ) input-output pairs  $\bar{X} = \{\bar{x}_m\}_{m=1}^M$  and  $\bar{f} = \{\bar{f}_m\}_{m=1}^M$ , which can be interpreted as “inducing points”. Consequently, these assumption leads to the data points likelihood as follows

$$p(y|x, \bar{X}, \bar{f}) = \mathcal{N}(y|K_{NM}K_M^{-1}\bar{f}, \Lambda + \sigma^2I), \tag{7}$$

Placing a Gaussian prior on the inducing targets  $p(\bar{f}|\bar{X}) = \mathcal{N}(\bar{f}|0, K_M)$ , we obtain the posterior distribution over the inducing targets

$$p(\bar{f}|D, \bar{X}) = \mathcal{N}(\bar{f}|K_M Q_M^{-1}(\Lambda + \sigma^2I)^{-1}y, K_M Q_M^{-1}K_M), \tag{8}$$

with  $Q_M = K_M + K_{MN}(\Lambda + \sigma^2I)^{-1}K_{NM}$ .

Thus, the predictions is given by marginalising out the inducing targets, which is a typical operation in Gaussian process. More specifically, Given a new data point, by integrating the likelihood Eq. (7) with the posterior Eq. (8), we have the distribution over the prediction

$$p(y_*|x_*, D, \bar{X}) = \int p(y_*|x_*, \bar{X}, \bar{f})p(\bar{f}|D, \bar{X})d\bar{f} \tag{9}$$

$$= \mathcal{N}(y_*|\mu_*, \sigma_*^2),$$

with the mean and covariance of prediction distribution given as

$$\mu_* = \mathbf{k}_*^T \mathbf{Q}_M^{-1} \mathbf{K}_{MN} (\Lambda + \sigma^2 \mathbf{I})^{-1} \mathbf{y} \tag{10}$$

$$\sigma_*^2 = K_{**} - \mathbf{k}_*^T (\mathbf{K}_M^{-1} - \mathbf{Q}_M^{-1}) \mathbf{k}_* + \sigma^2, \tag{11}$$

with  $K_{**}$  donating the covariance between new input points, also likewise for the other indexes. The inferring of the posterior distribution over the targets correspond to learn the regression model. In addition, with the assumption of connection of the inducing target and training data, the sparse Gaussian process can achieve effective learning with the computational complexity of  $O(NM^2)$ .

### 3.3 Rhythmic Movement as a CPG

Without feedback of the sensor data, Central Pattern Generators (CPGs) are defined as the neural circuits in the spinal cord of vertebrates. In this paper, the motion of each leg is address by each CPG. Each unit generator is driven by a non-linear function, which consists of a pattern generator defined by a rhythmic generator.

From Fig. 3, each leg is composed of 2 active DOFs, i.e. hip joint and knee joint both in the sagittal plane. The others

are passive joint as mentioned before. Besides, we describe the walking movement sequence by employing RMP. Each locomotion period starts with a new initial phase and maybe a different frequency. To regulate the desired phases of each exoskeleton leg, we couple among the neural oscillators. This motivation is driven by biological conception, which is naturally used in human locomotion and gait transition. The hypothesis assumes that dealing with neural oscillators is of great importance in desired phased coordinates [47].

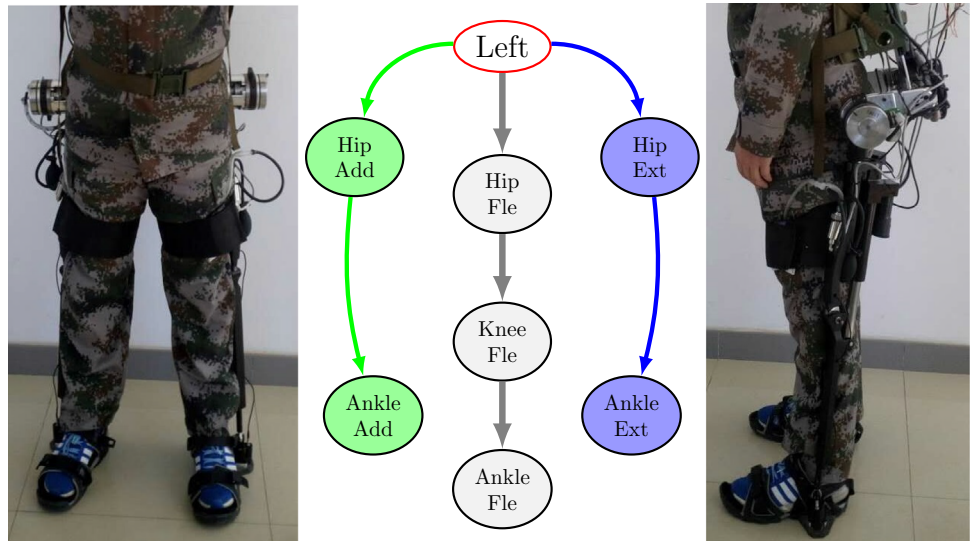
The period locomotion is implemented by a phase oscillator, and is also as a timer which can generate rhythmic movement for a single leg. The coupling terms of the oscillator  $i$  is introduced as follows

$$\dot{\phi}(t) = \omega + \kappa \sum_{i=1}^n C_{ij} \sin(\phi_j(t) - \phi_i(t) - \phi_{bias}), \tag{12}$$

where  $C_{ij}$  is a  $n \times n$  matrix, which presents the connection between the other oscillators, and  $\kappa$  is a positive coupling strength. Besides,  $\phi_i(t)$  and  $\phi_j(t)$  are reference trajectories, and  $\phi_{bias}$  is the phase bias between two reference phase trajectory.

Note that, the motivation behind this oscillator is that along with the rising of time keeping clock, the period phase monotonically increases with the rate  $\omega$ . The rhythmic signal of oscillator acts as a timer for the implementation of the rhythmic primitives. To be more specific, the formulation is applied to coordinate the desired phase connection. Among the canonical oscillators, we design the desired phase difference such that the joints in the same leg with zero phase difference, and the joints in the different leg with an opposite phase ( $\pi$  phase difference). More specifically, we define  $\phi_1, \phi_2$  as a joint phase of knee and hip in the left leg  $S_{left}$ , and  $\phi_3, \phi_4$  as a joint phase of knee and hip in the right leg  $S_{right}$ . Besides, the following condition is also required,

Fig. 3 General schematic of CPG locomotion



$\phi_1 - \phi_2 = 0, \phi_3 - \phi_4 = 0, \phi_1 - \phi_3 = \pi$  and  $\phi_2 - \phi_4 = \pi$ . Therefore, the element of relationship matrix  $C$  is written as below

$$C_{ij} = \begin{cases} 0 & \text{if } i = j; \\ 1 & \text{if } i, j \in S_{left} \text{ or } S_{right}; \\ -1 & \text{else.} \end{cases} \quad (13)$$

We employ the phase resetting to avoid the discontinuity of the motion planning. Figure 4 gives the design conception of exoskeleton shoes. The phase resetting is driven by the



**Fig. 4** Foot switch location. In order to maximize the sensitivity of the contact, the foot switches are mounted underneath ridges in the position of toe and heel respectively

specific event

$$\dot{\phi}_{ref} = \hat{\omega}_{ref}^n + \delta(t - t_{heel})(\phi_{heel} - \phi_{ref}) \quad (14)$$

$$\hat{\omega}_{ref}^{n+1} = \hat{\omega}_{ref}^n + K(\omega_{measured}^n - \hat{\omega}_{ref}^n). \quad (15)$$

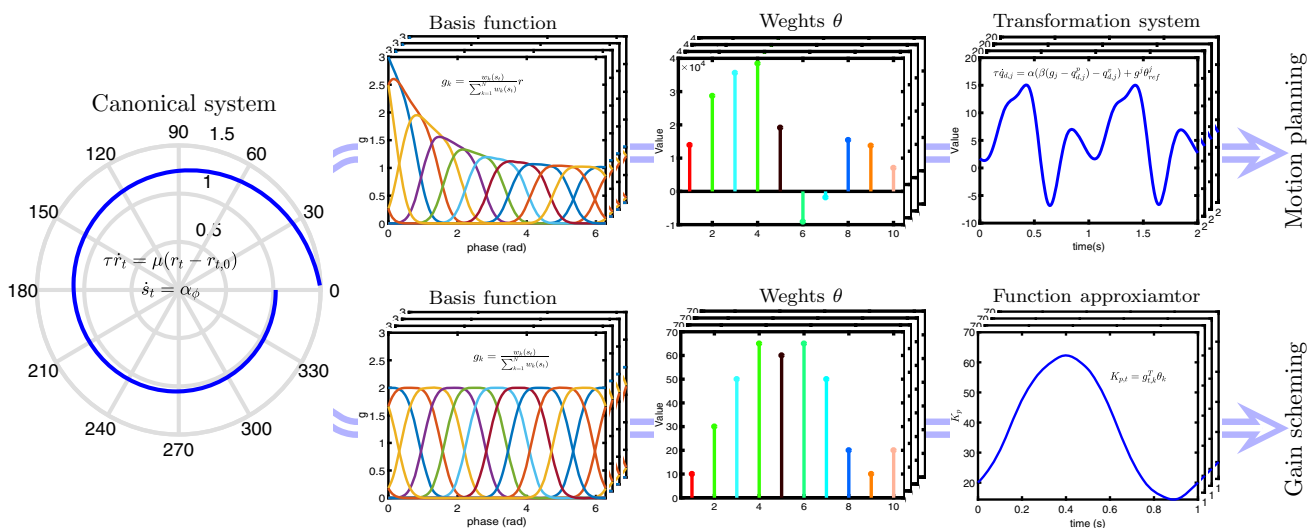
As soon as the heel of left leg strikes at the ground, the phase of the left leg  $\phi$  is defined as  $\phi_{heel} = 0$ , on the contrary, the phase of the right leg is defined as  $\phi_{heel} = \pi$  naturally.  $n$  is the steps' counter and  $\delta$  is known as Dirac delta function.

### 3.4 Motion Planning with Rhythmic Dynamical Movement Primitives

The desired state  $\dot{q}_{d,j}^p, \dot{q}_{d,j}^v$  are acquired by employing Rhythmic Movement Primitives (RMPs). Every RMP forms its trajectory, which means the following description hold for every DOF.

Two actuated joints (hip joints and knees joints) in each leg are equipped by a RMP. For the convenience, we define the indexes as follows:  $i = 1, i = 2$  for  $R_{hip}$  and  $R_{knee}$  respectively, and  $i = 3, i = 4$  for  $L_{hip}$  and  $L_{knee}$  separately. In this paper, we only present the necessary formulation, and we leave the more specific information to [41,48]. The one-DOF RMP is written as a state space model

$$\begin{bmatrix} \dot{q}_{d,j}^v \\ \dot{q}_{d,j}^p \end{bmatrix} = \frac{1}{\tau} \begin{bmatrix} \alpha\beta & \alpha \\ 0 & 1 \end{bmatrix} \begin{bmatrix} g - q_{d,j}^p \\ -q_{d,j}^v \end{bmatrix} + \frac{1}{\tau} \begin{bmatrix} g^j \theta^j_{ref} \\ 0 \end{bmatrix}, \quad (16)$$



**Fig. 5** Rhythmic movement primitives (RMPs). RMPs aim to generate arbitrary smooth continuous periodic movement primitives with a simple linear dynamical system as well as a nonlinear component  $g_k \theta$ . The nonlinear component is comprised of now von Mises basis functions, multiplied with a learning weight vector  $\theta$ . The canonical system

elements, amplitude  $r_t$  and phase  $s_t$  are presented in polar coordinates, essentially representing a periodic behaviour. The rhythmic movement has a period of 1 second with a reaching goal the end of one cycle. Although the proportional gain schedules  $K_p$  are not transformation system, we would rather define as a approximator  $K_{p,t} = g_k \theta$

where  $\alpha$  and  $\beta$  are positive constants and  $\tau$  is a time constant.  $\phi$  and  $r$  represents phase and amplitude respectively.  $g$  is defined as a goal. The second term  $g^j \theta_{ref}^j$  is a non-linear force term which is chosen to be periodic. The shape of movement is defined by the parameter  $\theta_{ref}^j$ , which is a learning parameter. The normal weight  $g^j$  is given by

$$g_k^j = \frac{w_k(s_t)}{\sum_{k=1}^N w_k(s_t)} r_j, \tag{17}$$

where  $w_k(s_t)$  is a non von Mise basis function, satisfying Eq. (18)

$$w_k(s_t) = \exp(h_k(\cos(\phi_j - c_k) - 1)). \tag{18}$$

However, the RMP system does not simultaneously converge to the goal. In order to couple with multiple DOFs exoskeleton locomotion in one dynamic system, a first-order linear dynamics so-called canonical system is presented in Eq. (19) as well as Eq. (20)

$$\tau \dot{r}_i = -\mu(r_i - r_{i,0}) \tag{19}$$

$$\dot{\phi}_i = \alpha_\phi, \tag{20}$$

where the canonical system is functioned with amplitude  $r_i$  and period phase  $\phi$ . Since the dependency of the explicit time is avoided, the dynamic system is an autonomous system.

As follows from Fig. 5, the core idea of RMPs is to generate a desired variables represented as position, velocity, and acceleration for the controller, then the controller transfers these variables to motor commands.

### 4 Incremental Human Machine Interaction Learning

Human walking locomotion consists of period movement sequences. Each sequence might differ with the others regarding time period, walking speed and locomotion profile. Therefore, it is difficult to propose a learning algorithm with one step of parameters updating that can perfectly describe walking sequences. In this subsection, we present an incremental reinforcement learning method and extend the sequence primitives to RMPs.

Our proposed learning methodology hinges on two contributions, i.e. subsection parameter updating of sequence primitives; subsection learning routine of Efficient  $PI^2$ -CMA-ES.

#### 4.1 Efficient $PI^2$ -CMA-ES

The  $PI^2$  algorithm is a reinforcement learning method, which is derived from the first principle of optimal control.

#### Algorithm 1 Efficient $PI^2$ -CMA-ES for single joint

```

Input:
1:  $\theta_{init}$ ; mean vector
2:  $A, A_{inv}$ ; Cholesky components and inverse
3:  $C$ ; penalty function
4:  $a_t = g_t^T \theta$ . parameterized policy
5:  $\bar{p}_{succ} = p_{succ}^{target}$ ; success rate
6:  $p_c = 0$ . evolution path
Output:
7:  $\theta_{final}$ ; learned mean
8:  $\Sigma_{final}$ . learned covariance
9:
10: repeat
11: for  $k = 0$  to  $K$  do
12:  $\epsilon \sim \mathcal{N}(0, \sigma^2 \Sigma)$  sampling
13:  $\tau_{k,i=1\dots N} \leftarrow \text{RMPs}(\theta_k)$  execute RMPs
14:  $C_{k,i=1\dots N} = C(\tau_{k,j})$  Equation (21)
15: end for
16: for  $i = 0$  to  $N$  do
17: for  $k = 0$  to  $K$  do
18:  $S_{k,i} = \sum_{j=i}^N C_{k,i}$  compute the costs
19:  $P_{k,i} = \frac{\exp(-\frac{1}{\lambda} S_{k,i})}{\sum_{k=1}^K \exp(-\frac{1}{\lambda} S_{k,i})}$  normalize weights
20: end for
21:  $\theta_i^{new} = \sum_{k=1}^K P_{k,i}(\theta_k + \epsilon_k)$  for all roll-outs
22: end for
23:  $\theta^{new} = \frac{\sum_{i=0}^N (N-i)\theta_i^{new}}{\sum_{l=0}^N (N-l)}$  for all phase steps
24: updateStepSize( $\sigma, \lambda_{succ}, \bar{p}_{succ}$ ) Algorithm 2
25: updateCholesky( $A, A_{inv}, \epsilon, \bar{p}_{succ}, p_c$ ) Algorithm 3
26: run one noiseless roll-out to evaluate the trajectory learning progress
27: until convergence
    
```

Its name comes from the Feynman–Kac lemma, which can be transformed from the Hamilton–Jacobi–Bellman lemma. It seeks to use parameterized policies from the point view of the probability-weighted averaging. Consequently, with the conclusion of the path integral stochastic optimal, the path cost for the specific RMPs case is given as

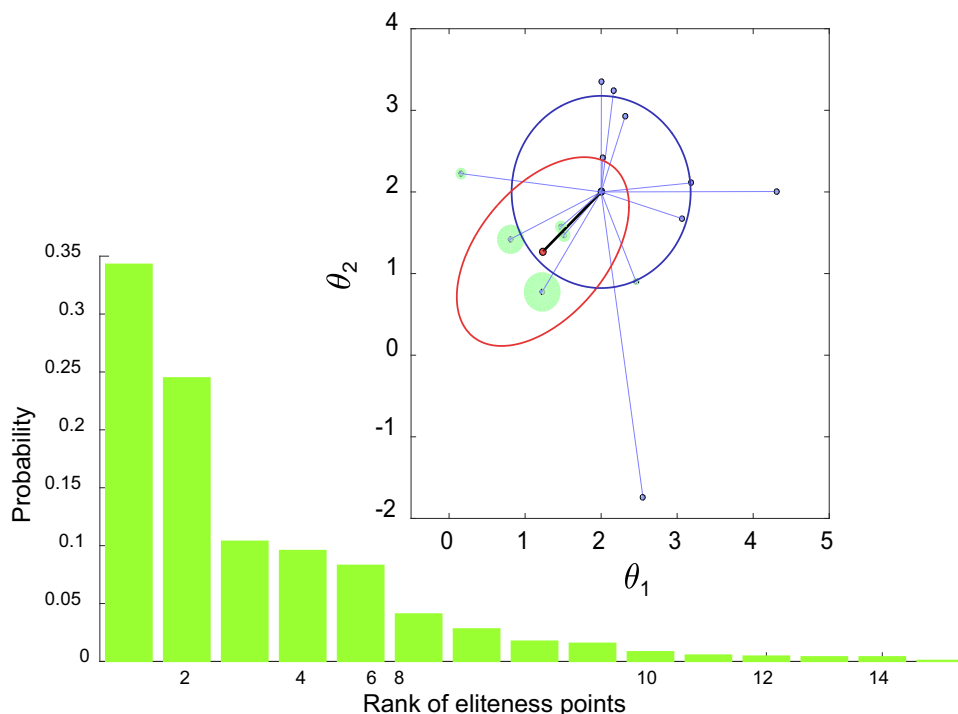
$$C(k, i) = \phi_{t_N,k} + \sum_{j=i}^{N-1} q_{t_j,k} + \frac{1}{2} \sum_{j=i+1}^{N-1} (\theta + M_{t_j,k} \epsilon_{t_j,k})^T R (\theta + M_{t_j,k} \epsilon_{t_j,k}), \tag{21}$$

where  $\phi_{t_N,k}$ ,  $q_{t_j,k}$  are the terminal cost and immediate cost respectively, and  $\epsilon_{t_j,k}$  is written as the samples from normal distribution. Besides,  $M_{t_j,k}$  is a projection matrix of the range space  $g_{t_j}$ , written as follows

$$M_{t_j,k} = \frac{R^{-1} g_{t_j,k} g_{t_j,k}^T}{g_{t_j,k}^T R^{-1} g_{t_j,k}}. \tag{22}$$



**Fig. 6** Upper right graph: the updated distributions of CMA-ES. Lower left graph: mapping from costs of sample points to probability. Note that the updated distributions do not consist of all the sample points, only the points with high probability are considered as eliteness



Furthermore, compared with the traditional reinforcement learning, the core idea is that the curse of the dimensionality associated with state action pairs can be simply avoided. By applying the constraints  $\Sigma = \lambda R^{-1}$ ,  $PI^2$  keeps a fixed update covariance. However this constraint is not necessary, since 1) positive-semid, the  $PI^2$  applies the finite constraint matrix can be seen as a vice versa and covariance matrix 2) a positive-semidefinite matrix with positive weight-average is also a positive-semidefinite matrix [49].

Rather than using a constant covariance matrix, we focus on a variant  $PI^2$  with an adaptation covariance matrix. The Covariance Matrix Adaptation Evolution Strategy (CMA-ES) [50] is developed for solving ill-conditioned problems by accelerating the convergence rate. Besides, CMA-ES can generate offspring candidates from parents and explore to the prevailing narrow valley. To be specific, the main difference of update rule is shown as in Fig. 6.

The main properties of CMA-ES used in this paper are introduced as follows:

1. The probabilities can be defined by the user and the default setting is fully explained by [50].
2. Samples are acquired from the Gaussian distribution  $N(\theta, \sigma^2 \Sigma)$ . Note that the covariance matrix is governed by the step size  $\sigma$  and shape  $\Sigma$ , and are updated respectively.
3. The covariance matrix and step-size are maintained in the ‘evolution path’, which seeks to update the parameters iteratively. The convergence speed is significantly

**Table 1** Default parameters for CMA evolution strategy

Step size parameters:

$$d = 1 + \frac{n}{2},$$

$$p_{succ}^{target} = \frac{2}{11},$$

$$c_p = \frac{1}{12}$$

Covariance adaptation:

$$c_c = \frac{2}{n+2}, c_{cov} = \frac{2}{n^2+6},$$

$$p_{thresh} = 0.44$$

improved since the correlations of the consecutive step is exploited incrementally.

The Efficient  $PI^2$ -CMA-ES proposed in this paper, is a novel variant of  $PI^2$  and CMA-ES, as presented in Algorithm 1. The interesting property is that it reduces the computing cost by Cholesky decomposition from  $O(n^3)$  to  $O(n^2)$  [51]. The main algorithm is introduced in three routines in pseudo-code and the default parameters are presented in Table 1. In the core part, the new candidate is selected from the normal distribution with the mean from a  $PI^2$  update rule and the covariance depending on the CMA-ES principle.

The indicator function is used to test the candidate elitism. Thus the value is one if the last mutation succeeds and otherwise zero.

The step size is updated relying on the learning rate  $c_p$  by a target default rate  $p_{succ}^{target}$ , as presented in Algorithm 2. The motivation is that the step size should be amplified if

---

**Algorithm 2** updateStepSize( $\sigma, \lambda_{succ}, \bar{p}_{succ}$ )

---

**Input:**

1:  $\sigma$ ; initial step size  
 2:  $\lambda_{succ}$ ; flag of indicator function  
 3:  $\bar{p}_{succ}$ ; evolution path

**Output:**

4:  $\sigma_{new}$ . new step size  
 5:   
 6:  $\bar{p}_{succ} \leftarrow (1 - c_p)\bar{p}_{succ} + c_p\lambda_{succ}$   
 7:  $\sigma_{new} \leftarrow \sigma \exp(\frac{1}{d} \frac{\bar{p}_{succ} - \bar{p}_{succ}^{target}}{1 - \bar{p}_{succ}})$  update step size

---



---

**Algorithm 3** updateCholesky( $A, A_{inv}, \epsilon, \bar{p}_{succ}, p_c$ )

---

**Input:**

1:  $A, A_{inv}$ ; Cholesky component and its inverse  
 2:  $\epsilon$ ; sampling noise  
 3:  $\bar{p}_{succ}$ ; success rate  
 4:  $p_c$ . evolution path

**Output:**  $A, A_{inv}$ . learned Cholesky component and its inverse

5:   
 6: **if**  $\bar{p}_{succ} > \bar{p}_{thresh}$  **then**  
 7:  $p_c \leftarrow (1 - c_c)p_c$  update evolution path  
 8:  $\alpha \leftarrow (1 - c_{cov} + c_{cov}c_c(2 - c_c))$   
 9: **else**  
 10:  $p_c \leftarrow (1 - c_c)p_c + \frac{1}{\sigma}\sqrt{c_c(2 - c_c)}A_{inv}\epsilon$   
 11:  $\alpha \leftarrow (1 - c_{cov})$   
 12: **end if**  
 13:  $\beta \leftarrow c_{cov}$  Cholesky component and its inverse  
 14:  $\omega \leftarrow A_{inv}p_c$   
 15:  $A \leftarrow \sqrt{\alpha}A + \frac{\sqrt{\alpha}}{\|\omega\|}(\sqrt{1 + \frac{\beta}{\alpha}\|\omega\|^2} - 1)p_c\omega^T$   
 16:  $A_{inv} \leftarrow \frac{1}{\sqrt{\alpha}}A_{inv} - \frac{1}{\sqrt{\alpha}\|\omega\|}(1 - \frac{1}{\sqrt{1 + \frac{\beta}{\alpha}\|\omega\|^2}})\omega[\omega^T A_{inv}]$

---

the success rate is a high value, from the point of heuristics, and the step size should be lessened since the success rate is low. Note that, the default parameters are rooted according to [52].

As described in Algorithm 3, instead of operation in covariance matrix, it can be directly implemented on the Cholesky components and their inverse. Thus, the update of the covariance matrix is never calculated explicitly.

### 4.2 Parameters Updating

In our proposed learning framework, the movement primitives and variable impedance gain are simultaneously

updated, as shown in Fig. 7. Thus, we rewrite the Eq. (16) and add a function approximator to  $K_{P,i}$

$$\frac{1}{\tau} \dot{q}_{d,j}^v = \alpha(\beta((g_j + \epsilon^g) - q_{d,j}^p) - q_{d,j}^v) + g_i^{i,T}(\theta_{ref}^i + \epsilon^i) \tag{23}$$

$$K_i = g_{i,K}^{i,T}(\theta_K^i + \epsilon_K^i). \tag{24}$$

Before implementing a roll-out, the shape exploration noise  $\epsilon^i$  and the gain exploration noise  $\epsilon_K^i$  are generated by the sampling from each Normal distribution respectively with mean  $\theta$  and covariance  $\Sigma$ .

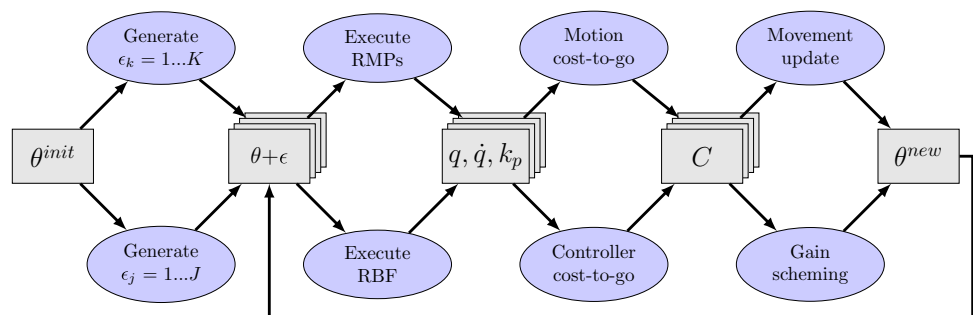
Then by these ‘noisy’ the RMP can generate a bunch of movement primitives  $[\dot{q}_{d,j}^p, \dot{q}_{d,j}^v]$  with a slight difference. Thus each trajectory leads to different penalties. According to these penalties, the parameters are updated by applying the proposed Efficient  $PI^2$ -CMA-ES. The most crucial part of this update rule is explained in the next subsection.

In Dynamic Movement Primitive (DMP) the  $g$  is written as a goal which leads the attractor to approximate the target, while in RMP the  $g$  is defined as a baseline. Considering the difference between each human walking cycle, the exoskeleton should adapt to the changeable movement primitives as long as the pilot’s path pattern varies. Thus the direct feedback derived from sensor observation data is the interaction force, which is changing during the human locomotion. An alternative explanation is that the trajectory does not fit for the pilot, and it should be adapted to the movement primitives. From intuition, we would rather have a changeable attracting goal than a fixed baseline. Besides, in order not to lose the advantage of the limit cycle attractor, the incremental goal learning is a good option.

Inspired by [53], we seek to update the goal and shape simultaneously. Note that, as for sequence primitives learning, the goal potentially influences not only the current trajectory penalty but also the whole sequence trajectories. When the learning of shape parameter is implemented, the goal should be kept constant so as not to avoid updating the shape of trajectory. Thus, there is no temporal dependency effect of goal on the cost.

To update the goal, the whole learning strategy remains the same. The main difference is that we update the goal

**Fig. 7** The generic loop of parameters updating. The policy improvement of gains and trajectory are updated separately, although they share the same cost functions



using the total trajectory cost, not a cost of single sequence movement. This means that the probability is only computed at the first start moment of the learning. Besides, in order to shape a smooth trajectory, we add another equation

$$\tau \dot{g} = \alpha_g (g_0 - g). \quad (25)$$

Accordingly, the goal parameters are updated on the basis of total trajectory cost, while the variable impedance parameters and shape parameters are updated according to every single sequence. Note that, each sequence is divided by walking resetting and the whole trajectory update consists of every sub-trajectory updates. The motivation behinds this is that the smoothness and continuity are considered with a simultaneous update of goal and trajectory parameters in order to avoid a jerk or stroke primitive.

## 5 Experiment Evaluation and Discussion

In this section, we conduct two experiments to evaluate the proposed methodology. The goal of the first experiment is to compare the learning performance of the proposed Efficient  $PI^2$ -CMA-ES with  $PI^2$  [48] and  $PI^{BB}$  [35]. Secondly, we compare the Efficient  $PI^2$ -CMA-ES with the Sensitivity Amplification Control (SAC) algorithm which is considered as a start-of-the-art control scheme typically for the exoskeleton [54].

### 5.1 Experiments on Single Joint Exoskeleton System

In order to evaluate the proposed learning algorithm, we apply  $PI^2$ ,  $PI^{BB}$  and our methodology on a single joint exoskeleton platform. In the experiment, the pilot's whole leg is attached with the exoskeleton, which makes the swing primitives possible only with the knee joint in the sagittal plane.

The  $PI^{BB}$  is a black-box optimization (BBO) algorithm, also derived from  $PI^2$ . The motivation behind choosing these two algorithms is that RL and BBO are two typical approaches to performing the optimization from action perturbation as well as parameter perturbation respectively. The policy improvement of BBO is focused on reward weighted averaging. However, the parameters updating is based on the gradient estimation. Thus, these two typical algorithms are the concrete implementation of the state-of-the-art RL and BBO.

For the following experiments, we define the learning task with the following immediate reward function, which is an implementation of a specific task according to [55]

$$r_t = W_{gain} \sum_j K_{P,t}^t + W_{acc} \ddot{x} + W_{via} f_{error}, \quad (26)$$

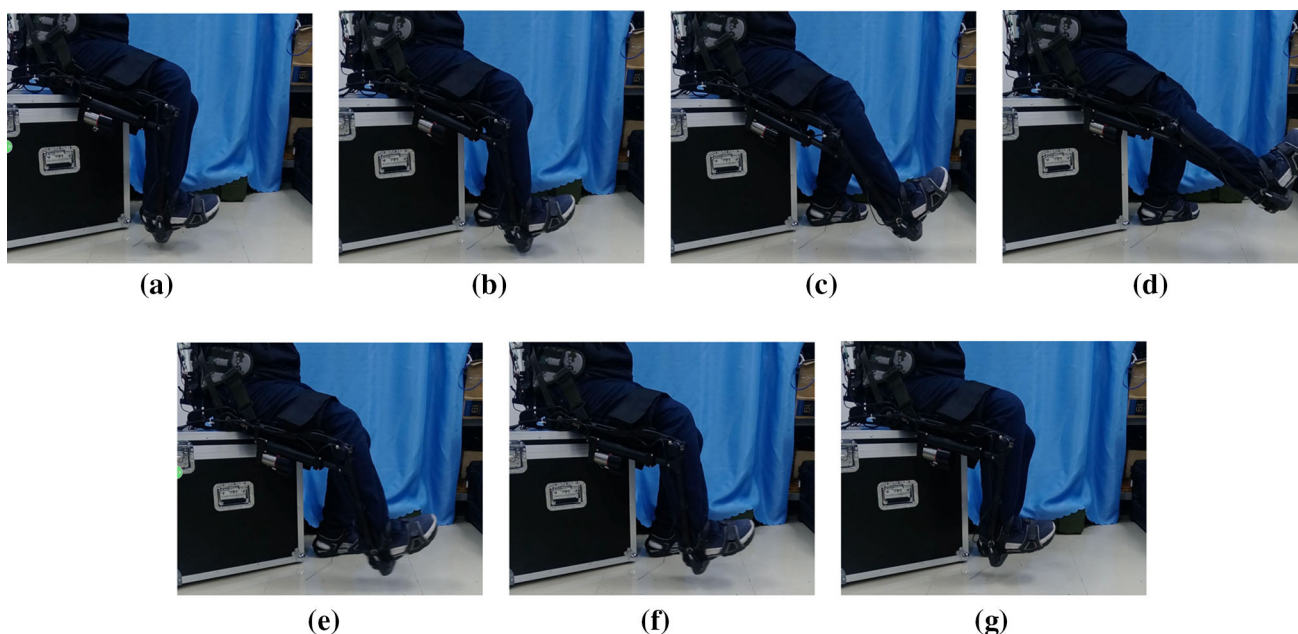
where  $\sum_j K_{P,t}^t$  is the sum of the control proportional gain over time. It seeks to improve desired performance such as energy saving, less wear and tear and the compliant interaction which leads to the behaviour of safety and robustness. The acceleration term  $\ddot{x}$  is to penalize the cost of high jerk of human motion. This regularization term corresponds to finding the trade-off between the generality of optimization as well as the fit to the learning trajectory. The term via-point  $f_{error}$  consists of several tracking errors and is written as follows

$$f_{error} = \sum_k \beta (q_k^d - q_k)^2, \quad (27)$$

where  $q_k$  is the joint angle planned by RMP, while  $q_k^d$  is the desired joint predicted with the SPGPs. Therefore, we do not need to mount sensors on the body of the pilot, such as sEMGs or IMU (inertial measurement units). Note that we choose several points from the trajectory to function the tracking error penalty, not all of the trajectory. The motivation behind this is that we would rather plan the movement primitives rather than follow the fixed trajectory.

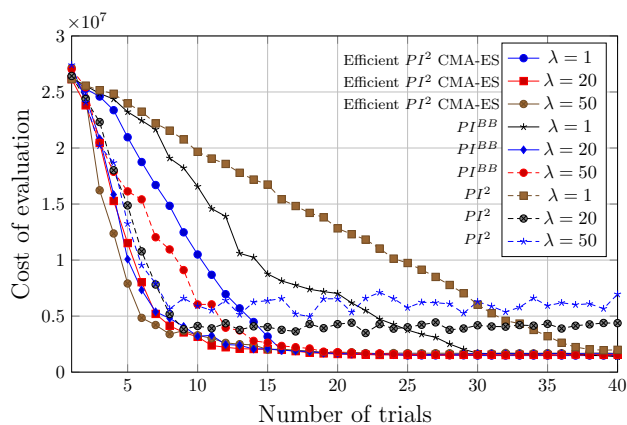
As shown in Fig. 8, the platform is driven by a DC motor, which provides extra strength for the pilot's knee joint. The encoder is mounted on the knee joint as well as the hip joint to measure the actual joint angle. The reference joint angle is provided with an inclinometer, which is attached to the pilot. In addition, the interaction between the pilot and the exoskeleton is compliant through a connection cuff. The pilot's shoes are attached to the exoskeleton shoes. Therefore, the pilot can interact with the exoskeleton flexibly and driven the exoskeleton easily with imposed force. Since the human locomotion is focused on the period walking pattern, the primitive of the knee joint is also designed as a periodic movement. The pilot is asked to sway his leg periodically, which is well interpreted in Fig. 8.

As can be seen from Figs. 9 and 10, the Efficient  $PI^2$ -CMA-ES outperforms better convergence behavior than  $PI^{BB}$  and  $PI^2$ . For  $PI^{BB}$  and Efficient  $PI^2$ -CMA-ES, the exploration magnitudes are varying over time with the update of covariance matrices. While for  $PI^2$ , the covariance matrices maintain the same during learning, since the covariance matrices are not updated. For  $\lambda = 20$ , both  $PI^{BB}$  and Efficient  $PI^2$ -CMA-ES have a nice convergence behaviour, while  $PI^2$  also shows a good behaviour in convergence, but “vibrates” with a high cost. Consequently, this kind of behaviour also appears when magnitude  $\lambda = 50$ , as  $PI^2$  has a larger penalty value. However, when  $\lambda = 1$  the convergence of  $PI^2$  is much slower, but performs better in convergence cost. Moreover, the Efficient  $PI^2$ -CMA-ES



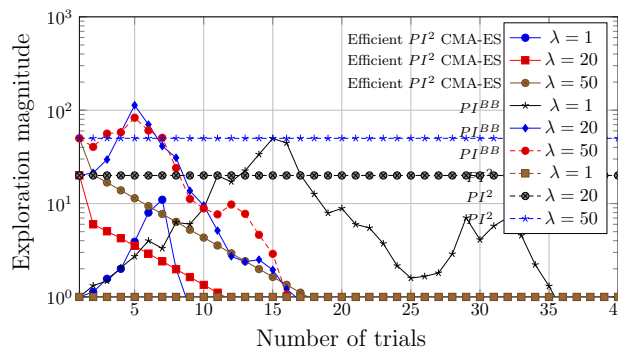
**Fig. 8** One DOF (knee joint) experiment. To illustrate our proposed learning scheme, the experiment is implemented on a single DOF exoskeleton platform. All active joint motors are turned off except the right knee to simulate as one DOF experiment platform. Also, through the connection cuff, the interaction between the pilot and the exoskeleton is compliantly connected and also the pilot’s shoes are attached to

the exoskeleton shoes. With the assistance torque from the DC motorized ball screw actuator, the pilot is asked to periodically sway his leg, which can be seen as a typical rhythmic movement primitive. From the a–g shown above, the periodic movements are recorded according to the simulation timer



**Fig. 9** Learning costs with different initial exploration magnitudes. We compare Efficient  $PI^2$ -CMA-ES,  $PI^{BB}$  and original  $PI^2$  with three different magnitudes ( $\lambda = 1, 10, 50$ )

converges at 18th step, barely influenced by the initial magnitude, while  $PI^{BB}$  converges at 30th step. The interesting property presented in these two figures is that a larger exploration magnitude does not always have a positive effect on learning speed. To be more specific, as shown in Fig. 9, for  $\lambda = 50$ , both  $PI^{BB}$  and  $PI^2$  present a poor behaviour in terms of convergence speed from step round 8th to step 18th, While  $PI^2$ -CMA-ES works better, converged at step 18th



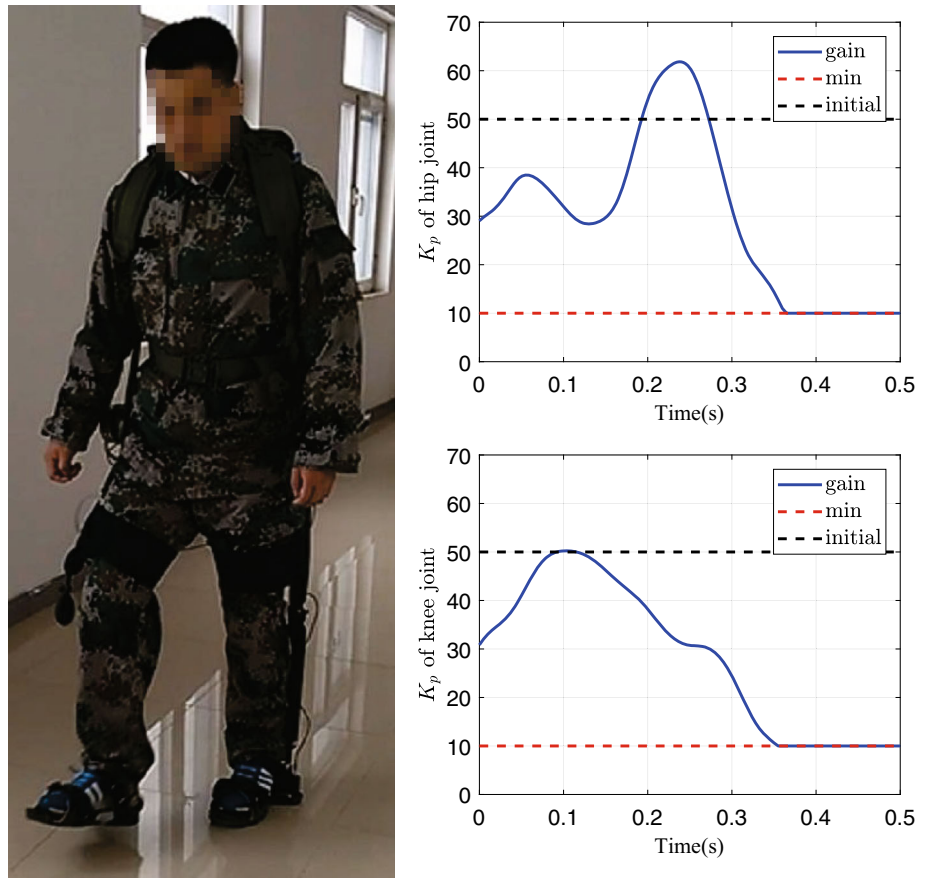
**Fig. 10** The exploration magnitudes during learning process. Note that both  $PI^{BB}$  and Efficient  $PI^2$ -CMA-ES update an adaptation covariance. However,  $PI^2$  keeps a fix covariance during learning procedure

nearly. It can be explained that although the larger magnitude is good for exploration, it can not work for the whole learning procedure. Besides, the variant covariance matrix is updated from the penalty which already exists.

From those results we have three conclusions for our proposed learning methodology: (1) The convergence speed of Efficient  $PI^2$ -CMA-ES does not rely on the initial value too much. However, for  $PI^2$  and  $PI^{BB}$ , a good initial magnitude start may have nice effect on learning speed. (2) The Efficient  $PI^2$ -CMA-ES increases exploration magnitude when



**Fig. 11** Left graph: flat walking experiment with LEAD system. Right upper graph: initial and final gain scheduling for hip joint of left leg. Right lower graph: initial and final gain scheduling for knee joint of left leg



the cost results in a high value. (3) The Efficient  $PI^2$ -CMA-ES automatically decreases exploration magnitude when the task has been fulfilled.

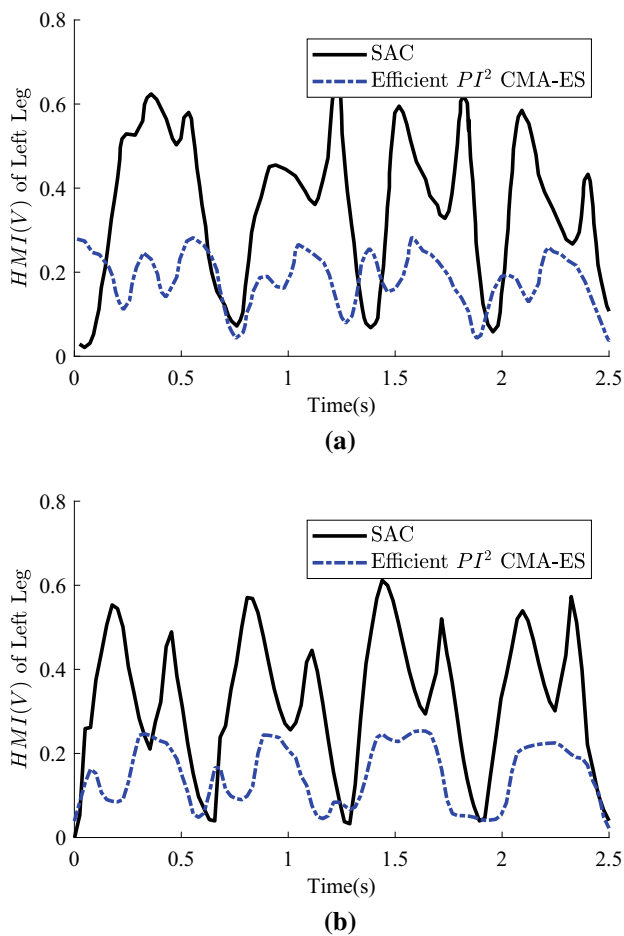
## 5.2 Experiments on the LEAD System

The develop LEAD control architecture mainly consists of the sensing system, the CAN (Controller Area Network) bus communication system and the embedded controller. The CAN bus communication system can achieve a high rate of 1Mbit per second real-time performance. The embedded coral controller is designed on dual-core processor Freescale and ARM-Cortex-A9 in an Ubuntu 12.04. The processing and fusion of sensor data, as well as the control command, is processed in this embedded board. The communication between the high-level controller and low-level driven system is connected to the first channel of the CAN, while the second channel is used to communicate with the sensor system. The sensor system is basically composed of the force switches, the force sensors, the optical encoders and incline angle system which is used to demonstrate the simulation of our proposed methodology.

For evaluating the control performance of our proposed learning methodology, we test our learning algorithm and

SAC control framework on the LEAD robotics system with a flat walking pattern, as shown in the left graph of Fig. 11. The SAC control algorithm is mainly designed to detect the human motion intention with the interaction between the exoskeleton and the pilot.

After the online learning in RMPs, the LEAD system shows a good performance in coping with the flat walking pattern. The gain self-tuning is presented in the right graph of the Fig. 11, which shows a learning procedure in a gait period. For gain scheduling, the initial gain for each joint is set to be 50, and also a lower bound is written as 10 according to experimental experience; After 40 learning loops, the gain scheduling is shown as the solid blue curve in the right graph of the Fig. 11. Besides, the Fig. 12 illustrates the control performance of the SAC algorithm and Efficient  $PI^2$ -CMA-ES methodology respectively. As shown in Fig. 12, comparing with the HMI force of SAC algorithm, the interaction force of Efficient  $PI^2$ -CMA-ES is much smaller in both right leg and right leg. Besides, the tracking error of the flat walking is shown in Fig. 14. The exoskeleton can follow the pilot's motion with little tracking error in flat walking, which illustrates that the HMI between the device and the pilot can be significantly reduced. Besides, since the movement primitives vary during the locomotion, through online learning of



**Fig. 12** Upper graph: right upper graph: the HMI comparison of right leg between SAC and efficient  $PI^2$ -CMA-ES. Lower graph: the HMI comparison of left leg between SAC and efficient  $PI^2$ -CMA-ES

parameters the exoskeleton can adapt to different walking primitives.

Furthermore, we extend experiments to going upstairs and downstairs, as shown in Fig. 13. The experimental results in Fig. 14 indicate that through incremental HMI learning the LEAD is also able to follow the more complex human motion and show a good control performance. Also, note that an interesting property in three walking patterns is that comparing the range of hip joint angle in three different walking pattern, we find that the range of hip motion in an upstairs case is larger than in flat walking. Moreover, when in downstairs case, the pilot lifts hip lower than in flat walking. From our intuition, we also feel more tired going upstairs the flat walking and downstairs.

## 6 Conclusion

In this paper, we propose a learning control methodology so-called Incremental Human–Machine Interaction Learn-

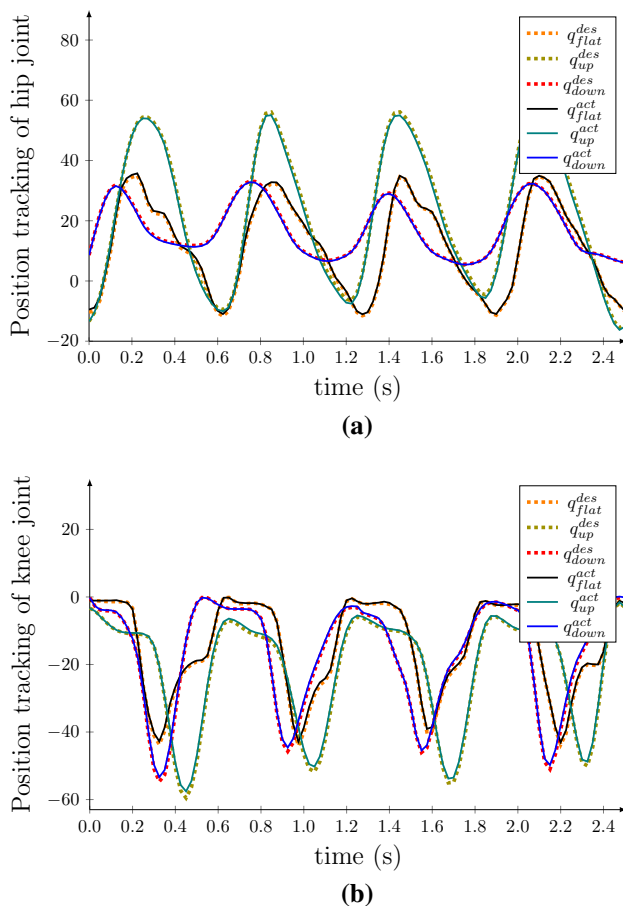


**Fig. 13** Left graph: the pilot with exoskeleton goes downstairs. Right graph: the pilot with exoskeleton goes upstairs. We extend the experiments to going upstairs and downstairs

ing for our Lower Extremity Augmentation Device. The reinforcement learning algorithm based on Efficient  $PI^2$ -CMA-ES can simultaneously learn rhythmic movement primitives and gain scheduling. Through incremental goal updating, the exoskeleton can adapt to different walking patterns by on-line trajectory updating.

Furthermore, in order to evaluate the performance and feasibility of our proposed learning methodology, we experiment on the single leg exoskeleton platform as well as the LEAD system. The results show a satisfying speed of learning convergence, compared with several state-of-the-art model-free reinforcement learning algorithms. Also, the HMI force between the pilot and the exoskeleton can be significantly reduced compared with the SAC control framework. Although the work done in this paper is mainly focused on the learning from human motion trajectory, the original design of this device is to enhance human endurance or augment human ability, such as weightlifting or hiking. However, if the user is a health human subject, the proposed algorithm in this paper may be a feasible solution since the human movement primitives is normal.

The future work will be focused on the learning of more complex movement primitives in more complicated constraint environment. The daily human life consists of many kinds of discrete and periodic motions. In addition, the environment is usually more unpredictable and stochastic than the flat ground or upstairs. Finally, understanding with limit sensor information is also a crucial task for human–machine coupling control.



**Fig. 14** Upper graph: tracking errors of hip joints of three different walking patterns (flat walking, upstairs, downstairs). Lower graph: tracking errors of knee joints of three different walking patterns (flat walking, upstairs, downstairs)

**Acknowledgements** We gratefully acknowledge the constructive comments and suggestions of the reviewers.

**Funding** Part of this work has received funding from National Natural Science Foundation of China under Grant No. 51521003

## Compliance with Ethical Standards

**Conflict of interest** The authors declare that they have no conflict of interest to disclose.

## References

- Cherry MS, Kota S, Young A, Ferris DP (2016) Running with an elastic lower limb exoskeleton. *J Appl Biomech* 32(3):269
- In H, Jeong U, Lee H, Cho KJ (2017) A novel slack enabling tendon drive that improves efficiency, size, and safety in soft wearable robots. *IEEE ASME Trans Mechatron* 22(1):59–70
- Ortiz J, Rocon E, Power V, de Eyto A, O’Sullivan L, Wirz M, Bauer C, Schülein S, Stadler K S, Mazzolai B, et al (2017) XoSoft—a vision for a soft modular lower limb exoskeleton. *Wearable robotics: challenges and trends*. Springer, Cham, pp 83–88

- Wang L, Du Z, Dong W, Shen Y, Zhao G (2018) Probabilistic sensitivity amplification control for lower extremity exoskeleton. *Appl Sci* 8(4):525
- Wang L, Du Z, Dong W, Shen Y, Zhao G (2018) Intrinsic sensing and evolving internal model control of compact elastic module for a lower extremity exoskeleton. *Sensors* 18(3):909
- Yu W, Rosen J, Li X (2011) PID admittance control for an upper limb exoskeleton. In: *Proceedings of the 2011 American control conference*, pp 1124–1129. IEEE
- Lee S, Sankai Y (2002) Power assist control for walking aid with HAL-3 based on EMG and impedance adjustment around knee joint. In: *IEEE/RSJ international conference on intelligent robots and systems*, 2002, vol. 2, pp 1499–1504. IEEE
- Tran HT, Cheng H, Duong MK, Zheng H (2014) Fuzzy-based impedance regulation for control of the coupled human-exoskeleton system. In: *2014 IEEE international conference on robotics and biomimetics (ROBIO)*, pp 986–992. IEEE
- Aguirre-Ollinger G, Colgate JE, Peshkin MA, Goswami A (2007) Active-impedance control of a lower-limb assistive exoskeleton. In: *2007 IEEE 10th international conference on rehabilitation robotics*, pp 188–195. IEEE
- Kazerooni H, Racine J L, Huang L, Steger R (2005) On the control of the Berkeley lower extremity exoskeleton (BLEEX). In: *Proceedings of the 2005 IEEE international conference on robotics and automation*, pp 4353–4360. IEEE
- Kazerooni H, Chu A, Steger R (2007) That which does not stabilize, will only make us stronger. *Int J Robot Res* 26(1):75
- Ghan J, Steger R, Kazerooni H (2006) Control and system identification for the Berkeley lower extremity exoskeleton (BLEEX). *Adv Robot* 20(9):989
- Mitrovic D, Klanke S, Howard M, Vijayakumar S (2010) Exploiting sensorimotor stochasticity for learning control of variable impedance actuators. In: *2010 10th IEEE-RAS international conference on humanoid robots*, pp 536–541. IEEE
- Siciliano B, Sciavicco L, Villani L, Oriolo G (2010) *Robotics: modelling, planning and control*. Springer, Berlin
- Stengel RF (2012) *Optimal control and estimation*. Courier Corporation, North Chelmsford
- Zhou K, Doyle JC, Glover K et al (1996) *Robust and optimal control*, vol 40. Prentice Hall, New Jersey
- Sutton RS, Barto AG (1998) *Reinforcement learning: an introduction*. MIT Press, Cambridge
- Huang R, Cheng H, Guo H, Chen Q, Lin X (2016) Hierarchical interactive learning for a human-powered augmentation lower exoskeleton. In: *2016 IEEE international conference on robotics and automation (ICRA)*, pp 257–263. IEEE
- Huang R, Cheng H, Guo H, Lin X, Zhang J (2017) Hierarchical learning control with physical human–exoskeleton interaction. *Information Sciences*, London
- Veneman JF, Kruidhof R, Hekman EE, Ekkelenkamp R, Van Asseldonk EH, Van Der Kooij H (2007) Design and evaluation of the LOPEX exoskeleton robot for interactive gait rehabilitation. *IEEE Trans Neural Syst Rehabil Eng* 15(3):379
- Hogan N (1984) Impedance control: an approach to manipulation. In: *American control conference*, 1984, pp 304–313. IEEE
- Hogan N (1985) Impedance control: an approach to manipulation: part II—implementation. *J Dyn Syst Meas Control* 107(1):8
- Huang H, Crouch DL, Liu M, Sawicki GS, Wang D (2016) A cyber expert system for auto-tuning powered prosthesis impedance control parameters. *Ann Biomed Eng* 44(5):1613
- Alquadi B, Modares H, Ranatunga I, Tousif SM, Lewis FL, Popa DO (2016) Model reference adaptive impedance control for physical human–robot interaction. *Control Theory Technol* 14(1):68
- Jacobson D, Mayne D (1970) *Differential dynamic programming*. Elsevier, Amsterdam

26. Lantoine G, Russell RP (2008) A hybrid differential dynamic programming algorithm for robust low-thrust optimization. In: AAS/AIAA astrodynamics specialist conference and exhibit, pp 152–173
27. Morimoto J, Atkeson CG (2003) Minimax differential dynamic programming: an application to robust biped walking. IEEE Press, London
28. Tassa Y, Erez T, Smart WD (2008) Receding horizon differential dynamic programming. In: Advances in neural information processing systems, pp 1465–1472
29. Li W, Todorov E (2004) Iterative linear quadratic regulator design for nonlinear biological movement systems. In: ICINCO(1), pp 222–229
30. Todorov E (2009) Efficient computation of optimal actions. Proc Nat Acad Sci 106(28):11478
31. Todorov E (2006) Linearly-solvable Markov decision problems. In: Advances in neural information processing systems, pp 1369–1376
32. Todorov E (2008) General duality between optimal control and estimation. In: 47th IEEE conference on decision and control, CDC, pp 4286–4292. IEEE
33. Williams RJ (1992) Simple statistical gradient-following algorithms for connectionist reinforcement learning. Mach Learn 8(3–4):229
34. Kober J, Peters JR (2009) Policy search for motor primitives in robotics. In: Advances in neural information processing systems, pp 849–856
35. Stulp F, Sigaud O (2012) Policy improvement methods: between black-box optimization and episodic reinforcement learning, p 34
36. Kopp C (2011) Exoskeletons for warriors of the future. Defence Today 9(2):38–40
37. Bucher D, Haspel G, Golowasch J, Nadim F (2000) Central pattern generators. eLS. Wiley Online Library
38. Ijspeert AJ (2008) Central pattern generators for locomotion control in animals and robots: a review. Neural Netw 21(4):642
39. Oliveira M, Matos V, Santos CP, Costa L (2013) Multi-objective parameter CPG optimization for gait generation of a biped robot. In: 2013 IEEE international conference on robotics and automation (ICRA), pp 3130–3135. IEEE
40. Matsubara T, Morimoto J, Nakanishi J, Sato M, Doya K (2005) Learning sensory feedback to CPG with policy gradient for biped locomotion. In: Proceedings of the 2005 IEEE international conference on robotics and automation, pp 4164–4169. IEEE
41. Ijspeert AJ, Nakanishi J, Hoffmann H, Pastor P, Schaal S (2013) Dynamical movement primitives: learning attractor models for motor behaviors. Neural Comput 25(2):328
42. Nakanishi J, Morimoto J, Endo G, Cheng G, Schaal S, Kawato M (2004) Learning from demonstration and adaptation of biped locomotion. Robot Auton Syst 47(2):79
43. Buchli J, Stulp F, Theodorou E, Schaal S (2011) Learning variable impedance control. Int J Robot Res 30(7):820
44. Stulp F, Buchli J, Ellmer A, Mistry M, Theodorou EA, Schaal S (2012) Model-free reinforcement learning of impedance control in stochastic environments. IEEE Trans Auton Ment Dev 4(4):330
45. Wolpert DM, Ghahramani Z, Jordan MI (1995) Are arm trajectories planned in kinematic or dynamic coordinates? An adaptation study. Exp Brain Res 103(3):460
46. Snelson E, Ghahramani Z (2006) Sparse Gaussian processes using pseudo-inputs. Adv Neural Inf Process Syst 18:1257
47. Ijspeert AJ (2001) A connectionist central pattern generator for the aquatic and terrestrial gaits of a simulated salamander. Biol Cybern 84(5):331
48. Theodorou E, Buchli J, Schaal S (2010) A generalized path integral control approach to reinforcement learning. J Mach Learn Res 11:3137
49. Stulp F, Sigaud O (2012) Path integral policy improvement with covariance matrix adaptation. arXiv preprint [arXiv:1206.4621](https://arxiv.org/abs/1206.4621)
50. Hansen N, Ostermeier A (2001) Completely derandomized self-adaptation in evolution strategies. Evol Comput 9(2):159
51. Igel C, Suttorp T, Hansen N (2006) A computational efficient covariance matrix update and a (1+1)-CMA for evolution strategies. In: Proceedings of the 8th annual conference on genetic and evolutionary computation, pp 453–460. ACM
52. Eigen M (1973) Ingo Rechenberg Evolutionsstrategie Optimierung technischer Systeme nach Prinzipien der biologischen Evolution. mit einem Nachwort von Manfred Eigen, Friedrich Frommann Verlag, Stuttgart-Bad Cannstatt
53. Stulp F, Schaal S (2011) Hierarchical reinforcement learning with movement primitives. In: 2011 11th IEEE-RAS international conference on humanoid robots (humanoids), pp 231–238. IEEE
54. Kazerooni H, Steger R, Huang L (2006) Hybrid control of the Berkeley lower extremity exoskeleton (BLEEX). Int J Robot Res 25(5–6):561
55. Theodorou E, Buchli J, Schaal S (2010) Reinforcement learning in high dimensional state spaces: a path integral approach. J Mach Learn Res 2010:3137

**Publisher's Note** Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

**Likun Wang** received his B.S. in Space Vehicles and Carrier Rockets from Bauman Moscow State Technology University in 2014, also in Russian Language and Literature, and in Space Vehicle Designing and Engineering from Harbin Institute of Technology at the same time. In 2015, he received his M.S. in Control Science and Engineering from Harbin Institute of Technology. He is currently working towards PhD degree in the State Key Laboratory of Robotics and System, Harbin Institute of Technology. His current research interests include exoskeleton, industrial robots and SLAM.

**Zhijiang Du** received his B.S., M.S. and Ph.D. degrees from Harbin Institute of Technology in 1995, 1997 and 2001, respectively. He is a professor with the State Key Laboratory of Robotics and System, Harbin Institute of Technology. His current research interests include medical robots, wearable robots, industrial robots and robots for special environments.

**Wei Dong** works with the State Key Laboratory of Robotics and System, Harbin Institute of Technology. His research expertise is in the general areas of robotics and mechatronics. He has extensive experience in a series of inter-related research subjects including innovative design of robot/mechatronics systems, robotic system modeling and optimization, and smart material and structure integration and application etc. Prior to joining HIT, he worked as a postdoctoral researcher at CNRS FEMTO-ST, France (2009.09–2010.08) and University of Connecticut, US (2007.11–2009.07), respectively. He received the Ph.D. and M.S. degrees in Mechatronics Engineering and B.S. degree in Mechanical Engineering from HIT in 2007, 2003 and 2001, respectively.



**Yi Shen** received his B.S., M.S. and Ph.D. degrees from the Harbin Institute of Technology in 1985, 1988 and 1995, respectively. He is a professor with the Department of Control Science and Engineering, Harbin Institute of Technology. His current research interests include fault diagnosis for control systems, flight vehicle control and ultrasound signal processing.

**Guangyu Zhao** received his B.S., M.S. in Computational Science and Engineering from the Jilin University in 2009 and 2012, respectively. He is a deputy researcher in Weapon Equipment Research Institute from China Ordnance Industries Group. His current research interests include wearable robots and embedded systems.