

Adaptive Gait Planning with Dynamic Movement Primitives for Walking Assistance Lower Exoskeleton in Uphill Slopes

Rui Huang,¹ Qian Wu,¹ Jing Qiu,^{2*} Hong Cheng,¹ Qiming Chen,¹ and Zhinan Peng¹

¹School of Automation Engineering, University of Electronic Science and Technology,
Chengdu 611731, China

²School of Mechanical and Electrical Engineering, University of Electronic Science and Technology,
Chengdu 611731, China

(Received August 1, 2019; accepted December 6, 2019)

Keywords: dynamic movement primitive, adaptive gait planning, lower limb exoskeleton, reinforcement learning

The lower exoskeleton system has attracted considerable interest in walking assistance of paraplegic patients. A critical issue in the walking assistance lower exoskeleton is how to generate gait motions for paraplegic patients. Predefined gait trajectory planning methods are widely used owing to their simplicity and effectiveness. However, a predefined gait trajectory planning method has three main drawbacks: (1) it requires a different gait model for different patients, (2) it cannot adapt to different terrains, such as slopes and stairs, (3) it does not consider the stability of the human exoskeleton system. In this study, we modeled the walking assistance lower exoskeleton with paraplegic patients as a human exoskeleton hybrid agent (HEHA). On the basis of the HEHA model, an adaptive gait planning method with dynamic movement primitives is proposed; in this method, the center of mass of HEHA is considered to ensure the stability of the human exoskeleton system. To adapt different pilots in slope scenarios, the reinforcement learning method is employed to update the parameters of the proposed gait model. The experimental results in both the simulation environment and the real-time exoskeleton system show that the proposed gait planning method makes the human exoskeleton system more stable in uphill slope scenarios.

1. Introduction

Lower limb exoskeletons have been widely used among strength augmentation, walking assistance, and rehabilitation-related scenarios. In walking assistance scenarios, lower limb exoskeletons are built for patients whose lower limbs are disabled, such as paraplegic patients. The purpose of these lower limb exoskeletons is to assist paraplegic patients in performing their daily activities. In the research on lower limb exoskeletons with paraplegic patients, a critical issue is how to rebuild the gait of the human exoskeleton system, especially for different patients in complex environments. For most lower limb exoskeleton systems for paraplegic patients, predefined gait trajectories, such as Rewalk,^(1–3) Ekso,^(4–7) and ATLAS, are employed.⁽⁸⁾

*Corresponding author: e-mail: qiuqing@uestc.edu.cn
<https://doi.org/10.18494/SAM.2020.2550>

In the predefined gait planning method, reference gait trajectories should be trained in different tasks and environments. For example, the lower exoskeleton should have different reference gait trajectories on different stairs and slopes. Moreover, the predefined gait planning method cannot adapt to different pilots.^(9,10) In the hybrid assistive limb (HAL) system,^(11–16) the myoelectric signal of the pilot's upper limb is employed to generate gait trajectories of the lower limb exoskeleton, which aim to adapt different pilots. In the proposed gait planning method in the HAL system, gait trajectories are generated on the basis of fusing the pilot's myoelectric signal, joint states, and plantar sensory information. However, in scenarios with complex terrains, this method is difficult to implement since the gait of the lower limb exoskeleton is related to the environment,⁽¹⁷⁾ such as slopes with different gradients.

In our previous work, we employed dynamic movement primitives (DMPs) for modeling gait trajectories of a lower limb exoskeleton with healthy pilots, which aim to adapt different gait trajectories in different walking speeds.^(18,19) Furthermore, the proposed gait models are employed for stair scenarios of the lower limb exoskeleton with paraplegic patients.⁽²⁰⁾ With the proposed gait model, the lower exoskeleton system could generate smooth gait trajectories by changing the goal position of the ankle joint.⁽²¹⁾ However, in the previous work, we did not consider the stability of the human exoskeleton system in different scenarios, as well as the comfort of the pilot during different tasks.

In this paper, we proposed an adaptive gait planning method based on DMP architecture and reinforcement learning, which aims to adapt different pilots in uphill slope scenarios. To obtain the center of mass (CoM) of the human exoskeleton system, we modeled the lower limb exoskeleton with a paraplegic patient as a human-exoskeleton hybrid agent (HEHA). Different from the previous gait model, the CoM of the human-exoskeleton system is employed in the proposed adaptive gait model. In the proposed adaptive gait model, goal states of the lower limb exoskeleton are calculated on the basis of the CoM of HEHA. For adapting different pilots, a reinforcement learning method based on policy improvement and path integrals (PI²) is employed to learn the parameters of the proposed gait model. In the reinforcement learning process, the stability performance of HEHA and the comfort of the pilot are both considered in the cost function. We validate the proposed gait planning method in the simulation environment, and experimental results show that the proposed gait method improves the stability of HEHA and the comfort of the pilot during walking in uphill slopes.

The structure of this paper is organized as follows. We generally introduce the proposed gait model based on the DMP architecture in Sect. 2. Afterwards, experimental results and discussion are given in Sect. 3. Finally, we conclude in Sect. 4.

2. Materials and Methods

In this section, we present the details of the proposed gait planning method in uphill slope scenarios. Firstly, we model the walking assistance lower exoskeleton with a paraplegic patient as HEHA in Sect. 2.1, in which the exoskeleton ankle trajectories are planned on the basis of the CoM of HEHA. Then, DMPs are employed to model the gait of the exoskeleton system, which describes ankle trajectories of the whole gait cycle. In Sect. 2.2, the gait learning process based

on the reinforcement learning method (PI²) is introduced, in which the stability of HEHA is considered in the cost function to achieve better stability performance.

2.1 Gait modelling with DMPs

In this section, we demonstrate the details of gait modeling in uphill slope scenarios for the lower exoskeleton system with paraplegic patients. Firstly, we modeled the human exoskeleton system as a hybrid agent, in which the gait phases of the human exoskeleton system are introduced. On the basis of the gait of HEHA, the goal position of ankle joints is utilized to plan the gait, in which the CoM of HEHA is embedded to ensure the stability of the human exoskeleton system. Finally, the DMP is utilized to model the ankle trajectories of the lower exoskeleton system, which aims to parameterize the gait model for a future learning process.

2.1.1 HEHA

In the lower exoskeleton system with paraplegic patients, the patient always needs to use crutches to maintain balance as well as to operate the exoskeleton.⁽²²⁾ In this study, we consider the lower exoskeleton system with the paraplegic patient in low walking speed, in which the patient should move sticks before the exoskeleton moves. Under this consideration, we model the human exoskeleton system as HEHA. In the HEHA model, the human exoskeleton system is regarded as a quadruped robotic system, in which two crutches are seen as two ‘front legs’ of HEHA, which are controlled by the patient.

In the framework of the proposed HEHA model, the lower exoskeleton system always follows the pilot’s crutches. In this paper, we separate four phases to describe a single step for the human exoskeleton system. Figure 1 shows four gait phases for a single step of HEHA.

Support Phase: The patient remains vertical with the support of the lower exoskeleton system and crutches [see Fig. 1(a)]. In this phase, the projection of the CoM of HEHA is located in the region of support, which is combined with the lower exoskeleton and crutches.

Transfer Phase: In this phase, the patient moves the crutches and his/her body to transfer the CoM of HEHA from the right leg to the left leg [see Fig. 1(b)].

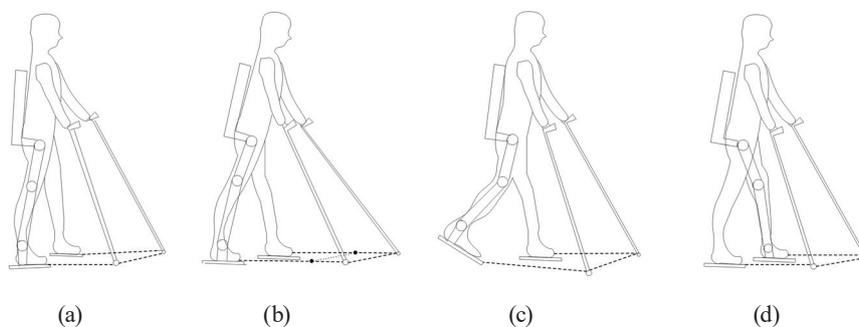


Fig. 1. Gait phases for a single step of HEHA: (a) support phase, (b) transfer phase, (c) swing phase, and (d) transition phase.

Swing Phase: The lower exoskeleton system moves the right leg to step forward with the patient [see Fig. 1(c)]. The goal position of the right leg will affect the walking speed of HEHA as well as the stability of the single step.

Transition Phase: Different from the transfer phase, the pilot moves the CoM of HEHA back to the center in the transition phase, which makes the human exoskeleton system return to the support phase [see Fig. 1(d)].

In the gait planning for the proposed HEHA, we focus on the goal position of the swing leg, in which planning a suitable goal position is the most important problem in this paper. Particularly in the uphill slope scenarios, the goal position of the swing leg will affect the stability performance of HEHA significantly.

2.1.2 Gait planning with the CoM in slope scenarios

During the walking process of the human exoskeleton system, the lower exoskeleton system should follow the movement of crutches to assist the patient. On the basis of the proposed HEHA, the gait of the lower exoskeleton system is planned to achieve a suitable goal position of the swing leg. In our previous work, we utilized the center of pressure (COP) to adjust the gait length of the human exoskeleton system during normal walking, which aims to achieve better stability performance for different patients.⁽²⁰⁾ However, in uphill slope scenarios, the human exoskeleton system should consider the information of HEHA in the sagittal plane. Hence, we utilize the CoM of HEHA in the vertical direction in this paper, which aims to adjust the goal position of the swing leg of the lower exoskeleton system.

Figure 2 shows two different step situations of the human exoskeleton system in slope scenarios. As shown in Fig. 2, the lower exoskeleton system should plan different goal positions of the swing leg when HEHA has different CoMs in the vertical direction. In the description in Fig. 2, Z_{com} represents the vertical distance in the z -direction of the CoM of HEHA, θ represents the degree of the slope, and (g_i, h_i) ($i = 1, 2$) represent the goal position of the planned gait, with g being the step length and h the step height.

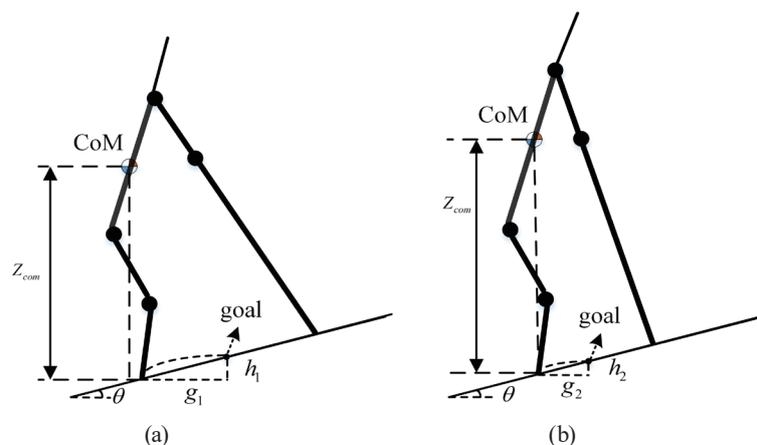


Fig. 2. Two different step situations of the human exoskeleton system in slope scenarios: (a) with higher CoM and (b) with lower CoM.

According to step situations described in Fig. 2, we build the model of the goal position based on the CoM of HEHA, which aims to ensure the stability of the human exoskeleton system. Equations (1) and (2) give the model of the goal position of the swing leg.

$$g = \frac{k}{Z_{com}} + t, \quad (1)$$

$$h = g \tan \theta, \quad (2)$$

where k and t are parameters for adjusting different goal positions for different patients, which need to be learned for the adaptation process. From the proposed model described in the above equations, the goal position could be calculated through the estimated CoM of HEHA. Since this paper is focused on the gait model and adaptation process, we will not introduce the estimation process for the CoM of HEHA.

2.1.3 Modelling with DMPs

With the calculated goal position based on the CoM of HEHA, another essential problem is how to generate joint trajectories for the lower exoskeleton system based on the goal position. In this paper, DMPs are utilized to model the trajectory of the ankle joint of the swing leg.^(23–25) Here, we ignore the inverse kinematics of the lower exoskeleton system, which transfers ankle trajectories to joint trajectories.

The DMP model is based on the nonlinear dynamical system, which describes a linear spring system perturbed by an external force. For a motion trajectory denoted by x , a DMP model can be defined as the following nonlinear system equation:

$$\tau \dot{v} = K(g - x) - Dv + (g - x_0)f, \quad (3)$$

where $v = \tau \dot{x}$ is an intermediate variable that indicates the first-order derivative of the output trajectory x , x_0 is the initial position of the system, g represents the goal position of the system, K and D are the stiffness and damping parameters of the system, respectively, and τ indicates the frequency of the system. f is a combination of nonlinear functions with Gaussian kernels:

$$f(s) = \frac{\sum_{i=1}^N \delta_i \psi_i(s)s}{\sum_{i=1}^N \psi_i(s)}, \quad (4)$$

$$\psi_i(s) = \exp(-h_i(s - c_i)^2), \quad (5)$$

where h_i and c_i are the height and width of Gaussian kernels, respectively. δ_i are weight parameters that can be obtained through the training process. In the initial training process of the DMP model, the nonlinear function f can be calculated on the basis of the input training trajectory:

$$f(s) = \frac{K(g-x) + Dv + \tau\dot{v}}{g-x_0}. \quad (6)$$

Here, the nonlinear function f is converted to the frequency domain for the convenience of calculations. After obtaining the nonlinear function f , the weight parameters δ_i are trained through the regression method.

With the trained DMP model with the input motion trajectory, the goal position of the DMP model can be set with different values to generate different trajectories. Figure 3 shows the change in goal position in the DMP model of ankle trajectories in the x - and z -directions. The simulation results show that the DMP model can generate ankle trajectories with different goal positions.

2.2 Gait adaptation with reinforcement learning

With the gait model described in Sect. 2.1, gait trajectories of the lower exoskeleton system can be generated through the DMP gait model with the CoM of HEHA. However, for different patients and slopes, the model described in Eq. (1) should be learned to obtain the optimal parameters of the proposed gait model. The learning process will be introduced in detail in this section.

To adapt different patients and slopes, a reinforcement learning method based on PI^2 is utilized to learn the parameters of the proposed gait model. Figure 4 shows the framework of the learning process of the proposed gait model. As shown in Fig. 4, on the basis of the DMP gait model with CoM, parameters in the gait planning model with CoM should be learned to adapt different patients. According to Eqs. (1) and (2), we set k and t as actions in the reinforcement learning process, which should be optimized in the learning process. The cost function is defined on the basis of improving the stability and comfort of the patient during walking on slope scenarios:

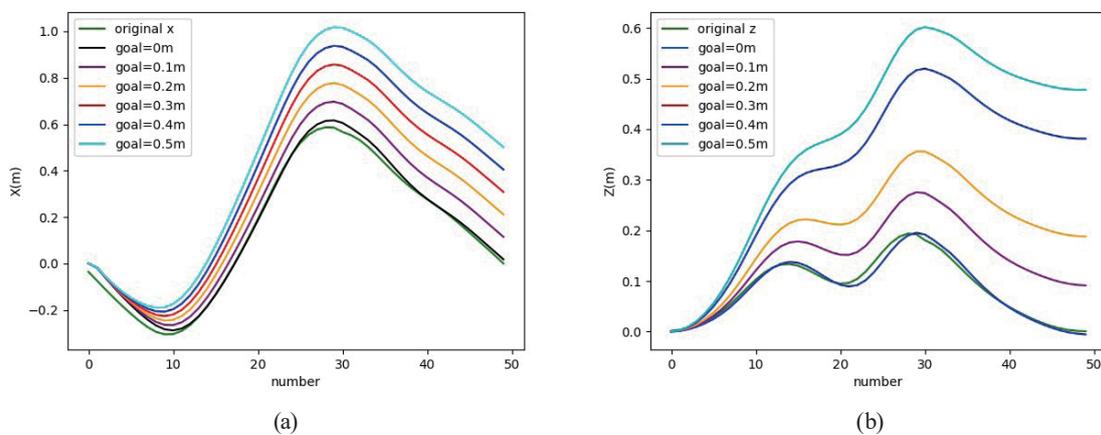


Fig. 3. (Color online) Changing goal positions with trained DMP model: (a) x -direction and (b) z -direction.

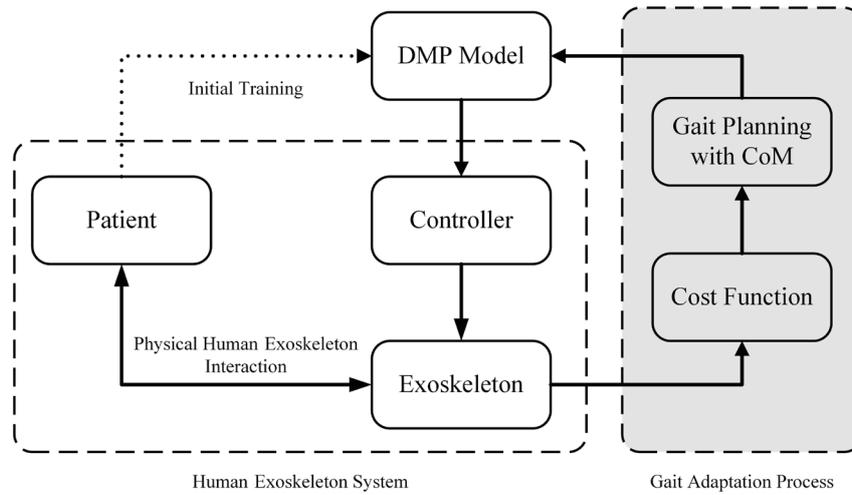


Fig. 4. Framework of the learning process of proposed gait model.

$$R = \omega_1(N - m) + \frac{\omega_2}{m} \sum_{i=1}^m r_i + \frac{\omega_3}{m} \sum_{i=1}^m Vp, \quad (7)$$

where N is the number of steps for each episode of the reinforcement learning process and m is the number of stable walking steps of each episode; the first term in Eq. (7) indicates the stability performance of HEHA in each episode. Vp is the variance of pitch angle of the upper body of HEHA in each episode, in which the pitch angle is measured by an inertial measurement unit (IMU) sensor installed on the upper body of the lower exoskeleton system. r_i is the average of the integral of ground reaction forces of crutches during each step:

$$r_i = \frac{1}{t} \int_0^t f dt, \quad (8)$$

where f indicates the ground reaction force, which is measured through plantar sensors on the crutches. The second term in Eq. (7) indicates the comfort of the patient in each episode. The weight parameters ω_1 , ω_2 , and ω_3 can be changed if we need to change the learning process for different terrains, which are set as fixed positive values in this paper.

With the defined cost function of the reinforcement learning process through Eq. (7), the learning process is described in Table 1. In the learning process, actions are set with k and t in Eq. (1), which are embedded in a vector $\theta = [k, t]^T$. δ_k is a Gaussian noise vector for action parameters, in which two elements of the noise vector are independent of each other. In the implementation of the PI² method, the model parameters are updated once every K times. During each episode, the path cost matrix S and probability matrix P need to be calculated through Eqs. (9) and (10):

$$S(k, i) = R_k + \frac{1}{2} \sum_{j=1}^N (\theta_k + \delta_k)^T (\theta_k + \delta_k), \quad (9)$$

Table 1

Parameter learning process of gait model by PI².Input: θ, δ, N, K **While** the cost function R does not reach the convergence condition in each episode:HEHA performs K times of random attemptsTest noise δ_k is added in each episodeFor $k \in [1, K]$:

$$\theta_k = \theta + \delta_k$$

For $i \in [1, N]$:For $k \in [1, K]$:Calculate the path cost matrix \mathbf{S} through Eq. (9)Calculate the probability matrix \mathbf{P} through Eq. (10)

End for

End for

For $i \in [1, N]$:Calculate the model parameter increments $\Delta\theta_i$ in each try by Eq. (11)Normalize model parameter increments $\Delta\theta$ by Eq. (12)Update model parameters $\theta = \theta + \Delta\theta$ Perform an evaluation experiment on the system with the parameter θ Calculate the cost function R **Output:** θ

$$P(k, i) = \frac{e^{-\frac{1}{\lambda} S(k, i)}}{\sum_{k=1}^K e^{-\frac{1}{\lambda} S(k, i)}}, \quad (10)$$

where k represents the k th walking test (from the first step to the step fall down), and j represents the j th step of walking in each walking test. R_k represents cost of the k th walking test. λ represents the discount factor, $\lambda \in (0, 1]$. Finally, the parameters of each step are calculated as

$$\Delta\theta_{t_i} = \sum_{k=1}^K [P(k, i) \delta_k]. \quad (11)$$

With the calculation of Eq. (11), model parameter updating should be normalized through Eq. (12), and finally the parameters of the proposed gait model should be updated.

$$\Delta\theta = \frac{\sum_{i=1}^{N-1} (N-i) \Delta\theta_{t_i}}{\sum_{i=1}^{N-1} (N-i)}, \quad (12)$$

$$\theta = \theta + \Delta\theta. \quad (13)$$

After each updating of model parameters, the cost function R should be calculated, which aims to determine if the parameters are optimal or not. If the cost function R has not reached the convergence condition, the learning process should be executed iteratively until we obtain the optimal parameters of the proposed gait model.

3. Experimental Results and Discussion

In this section, we give the details of the experimental results and discussion of the proposed gait model and its learning process in a simulation environment. In Sect. 3.1, we briefly introduce the simulation environment and the human exoskeleton model. Then, the reinforcement learning process of the proposed gait model is verified in the simulation environment, as well as in experiments for validating the learned gait model.

3.1 Simulation environment

In this paper, we validate the proposed gait model and its learning process in a simulation environment. The human exoskeleton model and simulation environment are based on Gazebo. Figure 5 shows the simulation environment with uphill slope scenarios. In the simulation environment, the human exoskeleton system is modeled as a special quadruped robot with a total of ten DOFs, in which eight of them are active. Four active degrees of freedom (DOFs) are set on hips and knees, and the others are set on the upper limbs for controlling crutches to simulate the interaction between the patient and the exoskeleton system.

3.2 Experiments and discussion

The simulation experiments are divided into two parts. The first part is the learning process of the proposed gait models, which learn the optimal parameters through the reinforcement learning process. The second part is validation with the learned gait models, which verify the proposed gait learning method in uphill scenarios.

In the parameter learning experiments, four cases are considered, in which four different initial values of the gait model are set. Table 2 shows the initial parameters of four cases in simulation experiments. This experiment aims to ensure the convergence of the reinforcement learning process and obtain the limits of initial parameters of the gait model. As shown in Table 2,

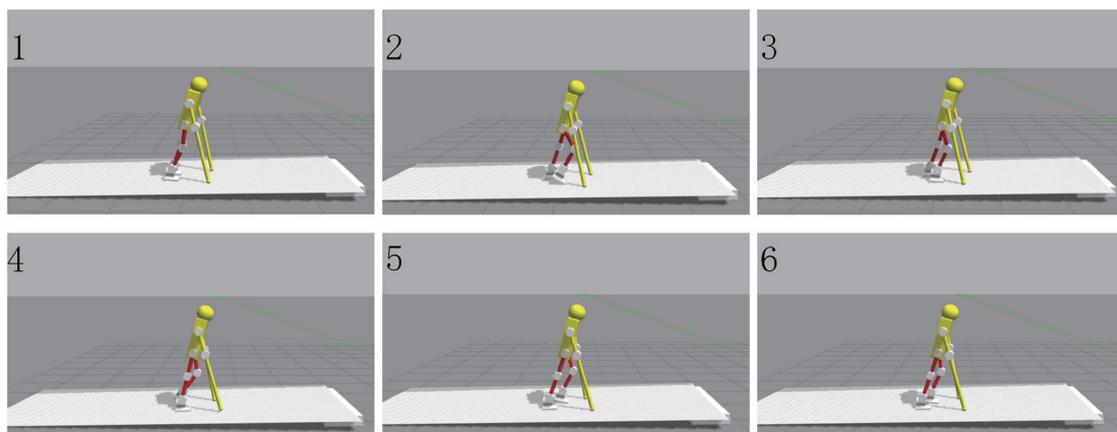


Fig. 5. (Color online) Process of HEHA's walking in uphill scenarios.

Table 2
Initial parameters in simulation experiments.

Initial parameters	k	t
Case 1	2	0.1
Case 2	1.74	0
Case 3	1.23	-0.2
Case 4	1	0.3

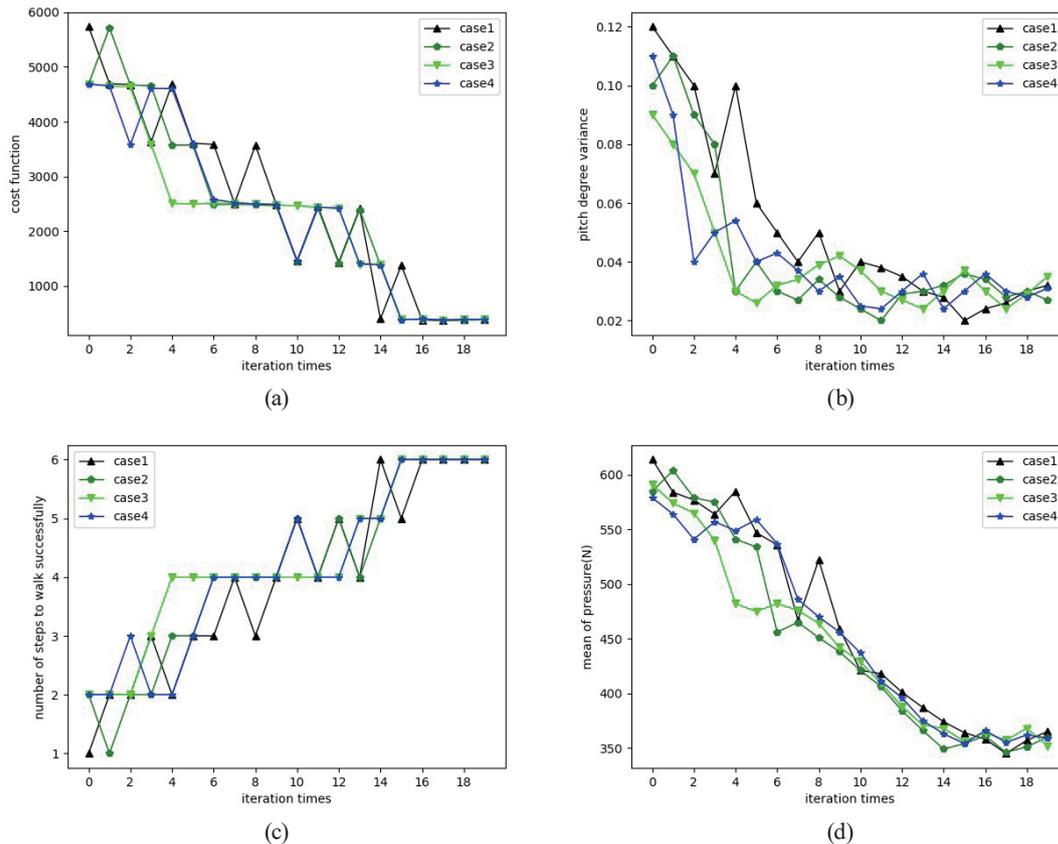


Fig. 6. (Color online) Experimental results of the learning process of model parameters: (a) convergence of the cost function, (b) pitch-angle variance of upper body of the human exoskeleton system, (c) number of stable walking steps, and (d) pressure on the crutches.

k is chosen in $[1, 2]$ and t is chosen in $[-0, 0.5]$. As a balance of stability and comfort of the human exoskeleton system, weight parameters in Eq. (7) are both chosen as $1/3$.

In the implementation of the reinforcement learning process, we set N as 6 and K as 5. Figure 6 shows the experimental results of the reinforcement learning process for gait parameter optimization. As shown in Fig. 6, the optimal model parameters can be learned within 20 iterations, with optimal parameters $[k, t] = [1.53, 0.13]$. Figure 6(a) shows the convergence of the cost function, which illustrates that different initial values of the parameters could achieve optimal results after the learning process. The pitch-angle variance of the upper body of the human exoskeleton system is shown in Fig. 6(b). Figure 6(c) shows the number of stable

walking steps during the learning process. As shown in Fig. 6(c), the human exoskeleton system can walk stably for all the N gait cycles. Figure 6(d) shows the pressure on the crutches during the learning process. The results show that after obtaining optimal gait parameters, the patient (upper limbs in the simulation model) could reduce the strength cost for stable walking (660 N compared with 360 N).

To compare the learned gait model with the traditional fixed step method, we validated the learned gait model with the human exoskeleton system on slope scenarios in the simulation environment. In the experiments, 6 fixed step lengths are chosen for comparison (from 0.10 to 0.15 m). Figure 7 shows comparisons of the proposed adaptive gait planning method and fixed step length methods. As shown in Fig. 7(a), the proposed adaptive gait planning method has reduced the pressure on the crutches significantly (almost 250 N compared with the 0.11 m fixed step length). With the comparison of the 0.15 m fixed step length, the pressure on the crutches gives almost the same performance. However, the 0.15 m fixed step length could not stabilize walking on the uphill slope scenarios. From Fig. 7(b), we can see that the proposed adaptive gait planning method can achieve stable walking in all the N gait cycles on the slope.

To evaluate the performance of the proposed adaptive gait planning method in different slope scenarios, experiments in different slopes with different gradients are carried out in the simulation environment. Table 3 shows experimental results of the human exoskeleton system in three different slopes (with 6, 12, and 18 degrees), with the comparison of the average pressure and the number of successful steps during walking on the slope. As shown in Table

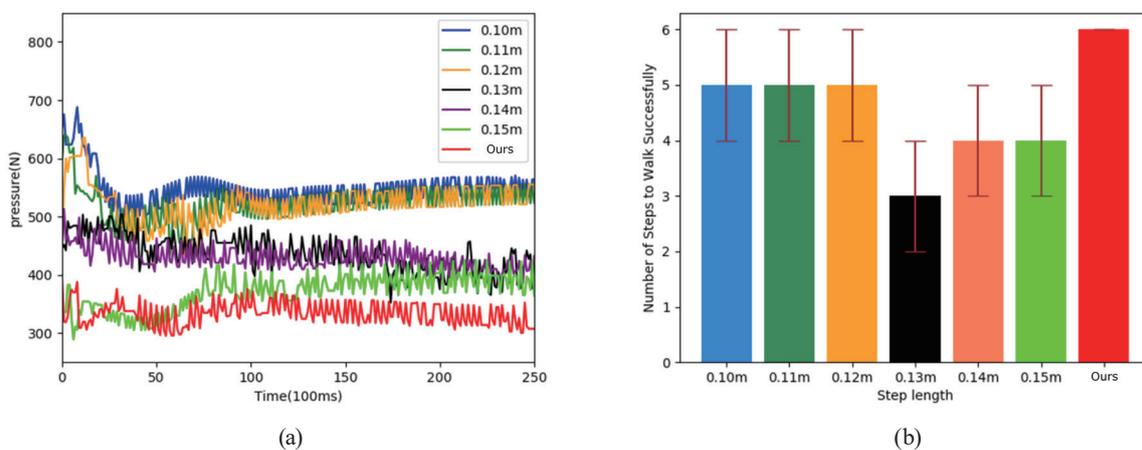


Fig. 7. (Color online) Comparison of the proposed adaptive gait planning method and fixed step length methods: (a) pressure on the crutches and (b) number of stable walking steps.

Table 3
Experimental results in different slopes with different gradients.

Slopes	Average pressure (N)	Number of steps
6 deg gradient	283.4	10
12 deg gradient	325.2	10
18 deg gradient	408.7	9

3, we can see that in the human exoskeleton system in slopes with larger gradients, the patient should experience more pressure on the crutches during the whole experiment. Experiments in the simulation environment show that the proposed adaptive gait planning method can achieve better performance than the fixed step length methods, with the improvement of the stability characteristic, as well as reducing the pressure on the crutches significantly to give a comfortable physical interaction for the patient.

4. Conclusions and Future Work

In this paper, we presented an adaptive gait planning method for a walking assistance lower exoskeleton in uphill slopes. The proposed gait planning method considers a gait model based on CoM and utilizes reinforcement learning to learn the optimal parameters of the gait model according to the stability and comfort performance. Experimental results show that the proposed method can improve the stability and comfort of patients during walking in uphill slopes.

In the future, we will extend our gait planning method to more application scenarios, such as complex outdoor environments with different terrains. Furthermore, more evaluation experiments should be carried out in real time with the lower exoskeleton system.

Acknowledgments

This work was made possible with support from the National Key Research and Development Program of China (No. 2017YFB1302300) and NSFC (Nos. 6150020696 and 61503060).

References

- 1 A. Esquenazi, M. Talaty, A. Packel, and M. Saulino: *Am. J. Phys. Med. Rehabil.* **91** (2012) 911. <https://doi.org/10.1097/PHM.0b013e318269d9a3>
- 2 G. Zeilig, H. Weingarden, M. Zwecker, I. Dudkiewicz, A. Bloch, and A. Esquenazi: *J. Spinal. Cord. Med.* **35** (2012) 96. <https://doi.org/10.1179/2045772312Y.0000000003>
- 3 K. Raab, K. Krakow, F. Tripp, and M. Jung: *Spinal. Cord. Ser. Cases.* **2** (2016) 15025. <https://doi.org/10.1038/scsanc.2015.25>
- 4 J. Pransky: *Ind. Robot.* **41** (2014) 329. <https://doi.org/10.1108/Ir-05-2014-0334>
- 5 K. A. Strausser and H. Kazerooni: *IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IEEE, 2011)* 4911. <https://doi.org/10.1109/iros.2011.6048674>
- 6 R. Steger, S. H. Kim, and H. Kazerooni: *IEEE Int. Conf. Robotics and Automation (IEEE, 2006)* 3469. <https://doi.org/10.1109/Robot.2006.1642232>
- 7 A. Zoss, H. Kazerooni, and A. Chu: *IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IEEE, 2005)* 3132. <https://doi.org/10.1109/Iros.2005.1545453>
- 8 D. Sanz-Merodio, M. Cestari, J. Conf. Arevalo, and E. Garcia: *Int. J. Adv. Robot. Syst.* **9** (2012) 1692. <https://doi.org/10.1109/Art.23210.5772/51903>
- 9 C. Yang, C. Zeng, Y. Cong, N. Wang, and M. Wang: *IEEE Trans. Ind. Inform.* **15** (2018) 1153. <https://doi.org/10.1109/TII.2018.2826064>
- 10 C. Yang, C. Zeng, C. Fang, W. He, and Z. Li: *IEEE/ASME Trans. Mech.* **23** (2018) 1193. <https://doi.org/10.1109/TMECH.2018.2817589>
- 11 H. Kawamoto, S. Kanbe, and Y. Sankai: *Ro-Man 2003: IEEE Int. Workshop on Robot and Human Interactive Communication (IEEE, 2003)* 67. <https://doi.org/10.1109/roman.2003.1251800>

- 12 S. Lee and Y. Sankai: IEEE Trans. Syst. Man Cybern. (2003) 1642. <https://doi.org/10.1109/icsmc.2003.1244648>
- 13 H. Kawamoto, S. Lee, S. Kanbe, and Y. Sankai: IEEE Trans. Syst. Man Cybern. (2003) 1648. <https://doi.org/10.1109/icsmc.2003.1244649>
- 14 S. Lee and Y. Sankai: IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IEEE, 2002) 1499. <https://doi.org/10.1109/irids.2002.1043967>
- 15 M. Hassan, H. Kadone, K. Suzuki, and Y. Sankai: Sensors **14** (2014) 1705. <https://doi.org/10.3390/s140101705>
- 16 M. Hassan, H. Kadone, K. Suzuki, and Y. Sankai: IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IEEE, 2012) 1609. <https://doi.org/10.1109/iros.2012.6386248>
- 17 C. Yang, C. Zeng, P. Liang, R. Li, and C. Y. Su: IEEE Trans. Autom. Sci. Eng. **15** (2018) 340. <https://doi.org/10.1109/TASE.2017.2743000>
- 18 R. Huang, H. Cheng, H. L. Guo, X. C. Lin, and J. W. Zhang: Inf. Sci. **432** (2018) 584. <https://doi.org/10.1016/j.ins.2017.09.068>
- 19 R. Huang, H. Cheng, J. Qiu, and J. W. Zhang: IEEE Trans. Autom. Sci. Eng. **16** (2019) 1566. <https://doi.org/10.1109/TASE.2018.2886376>
- 20 Q. M. Chen, C. B. Zhou 2018: Ind. Robot. **46** (2019) 421. <https://doi.org/10.1108/IR-03-2018-0054>
- 21 W. H. Ma, H. Cheng, R. Huang, and Q. M. Chen: IEEE Int. Conf. Biomedical Robotics and Biomechatronics (IEEE, 2018) 708. <https://doi.org/doi:10.1109/robio.2018.8664881>
- 22 K. A. Strausser, T. A. Swift, A. B. Zoss, H. Kazerooni, and B. C. Bennett: IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IEEE, 2011). <https://doi.org/10.1109/IROS.2011.6095025>
- 23 A. J. Ijspeert, J. Nakanishi, and S. Schaal: IEEE Int. Conf. Robotics and Automation (IEEE, 2002) 1398. <https://doi.org/10.1109/ROBOT.2002.1014739>
- 24 A. J. Ijspeert, J. Nakanishi, and S. Schaal: IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IEEE, 2002) 958. <https://doi.org/10.1109/IROS.2002.1041514>
- 25 A. J. Ijspeert, J. Nakanishi, and S. Schaal: IEEE/RSJ Int. Conf. Intelligent Robots and Systems (IEEE, 2001) 752. <https://doi.org/10.1109/IROS.2001.976259>