Robotica (2022), **40**, pp. 2732–2747
doi:10.1017/S0263574721001934

CAMBRIDGE
UNIVERSITY PRESS

RESEARCH ARTICLE

Motion generation for walking exoskeleton robot using multiple dynamic movement primitives sequences combined with reinforcement learning

Peng Zhang and Junxia Zhang*

Tianjin University of Science and Technology, Dagunan Road, Tianjin, China and Tianjin Key Laboratory for Integrated Design and Online Monitor Center of Light Design and Food Engineering Machinery Equipment, Tianjin, China
*Corresponding author. E-mail: zjx@tust.edu.cn

Received: 6 September 2021; **Revised:** 25 November 2021; **Accepted:** 10 December 2021;
First published online: 7 January 2022

Keywords: dynamic movement primitives, walking exoskeleton robot, gait planning strategy, reinforcement learning

Abstract

In order to assist patients with lower limb disabilities in normal walking, a new trajectory learning scheme of limb exoskeleton robot based on dynamic movement primitives (DMP) combined with reinforcement learning (RL) was proposed. The developed exoskeleton robot has six degrees of freedom (DOFs). The hip and knee of each artificial leg can provide two electric-powered DOFs for flexion/extension. And two passive-installed DOFs of the ankle were used to achieve the motion of inversion/eversion and plantarflexion/dorsiflexion. The five-point segmented gait planning strategy is proposed to generate gait trajectories. The gait Zero Moment Point stability margin is used as a parameter to construct a stability criteria to ensure the stability of human-exoskeleton system. Based on the segmented gait trajectory planning formation strategy, the multiple-DMP sequences were proposed to model the generation trajectories. Meanwhile, in order to eliminate the effect of uncertainties in joint space, the RL was adopted to learn the trajectories. The experiment demonstrated that the proposed scheme can effectively remove interferences and uncertainties.

1. Introduction

The exoskeleton system was mainly designed for users with muscle injury to enhance motion ability in daily activities. In the early stage of recovery, users keep completely passive and the exoskeleton provides support and guidance to ensure walking along the desired trajectory, and users tend to perform exercises with reduced muscle activity and metabolism. Exoskeleton robot is a highly human-machine integrated technology, which is mainly used for rehabilitation and assistance operations for the disabled and the aged. Exoskeleton robots have been discussed since the 1960s. In 1970, the United States produced the first exoskeleton robot system (Hardman). In 2000, the US Defense Advanced Research Projects Agency (DARPA) launched the Enhanced Human Exoskeleton Program (EHPA), which produced many remarkable achievements, such as BLEEX (2004), ExoHiker (2005), ExoClimber (2005), HULC (2009), and XOS2 (2010) [1–6]. In addition, other research units have published research results in the field of exoskeleton robots, such as Japan's "HAL" (2008) [7], Israel's Rewalk (2010) [8]. However, the poor experience caused by the mismatch of human-machine motion trajectory has become a bottleneck for the application and promotion of lower limb exoskeleton robots, which is also a common problem in the field of lower limb exoskeleton robots.

The lower extremity exoskeleton robot has many degrees of freedom (DOFs) and complex structure, so a reasonable gait planning method is necessary to achieve stable and efficient walking. At present, the gait planning methods include the bionics gait planning [9–11], the modeled gait planning [12, 13], and the intelligent algorithm gait planning [14–18]. Honda has developed a gait algorithm for ASIMO based on bionics gait planning [9]. However, huge amounts of data are needed to be collected by sensors, which caused tremendous calculation. According to the structural characteristics of the robot, simplified models were established to conduct gait planning, such as the multiple-link model [12] and inverted pendulum model [13]. But this method can only be used for simple motion scenes. The gait planning method based on an intelligent algorithm is calculated by a certain number of intermediate processing units. The generalized coordinates and the velocity of each joint in each walking cycle are taken as the input node variables. The joint angle and driving torque were output [16, 17]. The neural network is used to adjust the parameters, such as step size and speed. The disadvantages of this method are complex calculation, high cost, and heavy workload. The idea of encoding motion through a dynamical system has been widely accepted. The complex motion of a robot can be considered as a combination of a series of simple elementary actions. Stefan Schaal's laboratory firstly proposed the dynamic movement primitives (DMP) method in 2002 [19]. It is not only suitable for trajectory planning of point-to-point motion mode but also suitable for periodic motion [20]. And DMP could be designed as building blocks to generate more complex motions through real-time sorting and modulation. Reference paper [21] suggests to use the framework of DMP and stochastic optimal control to derive a novel trajectory planning approach. And a learning experiment on a simulated 12 degree-of-freedom robot dog illustrates the functionality of the algorithm in a complex robot learning scenario. Reference paper [22] presents a novel planning strategy based on DMP, which is applicable to high performance unmanned aerial vehicles. However, none of them are related to the control and dynamic programming of the man-machine system. With the goal to generate more scalable algorithms with higher efficiency and fewer open parameters, RL algorithm has recently moved towards combining classical techniques from optimal control and dynamic programming with modern learning techniques from statistical estimation theory. The method of strategy improvement was realized through trial and error and environment interaction. It has the ability of self-learning and online learning [23–30]. A popular reinforcement learning model is the Markov decision process (MDP) model, which was used for discrete or random problems [25]. However, it was not appropriate for high-dimensional continuous dynamic systems, which was not easy to be iterated. Theodorou suggests to use the framework of stochastic optimal control with path integrals to derive a novel approach to RL with parameterized policies. A learning experiment on a simulated 12 degree-of-freedom robot dog illustrates the functionality of the algorithm in a complex robot learning scenario [28]. The function value approximation method of stochastic Hamilton Jacobi Bellman [31] method and the direct strategy learning method based on path integrals are applied. The problem of statistical reasoning is solved through the continuous learning and training of samples.

Although the trajectory learning method based on DMP has achieved a lot of achievements, the method still has some problems. When the initial value and the target value are consistent, the learning trajectory is a straight line. The trend of learning trajectory is not similar to that of demonstration teaching, and the method of dynamic motion element is invalid. This paper describes a novel coupled movement planning and adaption based on DMP and RL algorithms for lower exoskeleton robots. The five-point segmented gait planning strategy is proposed to generate gait trajectories. The novel multiple-DMP sequences were proposed to model the joint trajectories. However, DMP is insensitive to perturbations. By exploiting the RL algorithm, the exoskeleton system could overcome interference and can learn the given motion trajectory. Until now, to the author's best knowledge, there is no work investigating movement sequences planning in-depth for walking exoskeleton.

2. Exoskeleton robot system

The developed lower-limb exoskeleton is shown in Fig. 1. The exoskeleton robot designed in this paper mainly considers the motion in the sagittal plane. For each leg, it has one active DOF for the hip joint and knee joint, respectively, and one passive DOF for the ankle joint.

According to the normal walking data measured by the gait experiment, the flexion limit of the hip joint is 40° and the extension limit is 35°. The flexion limit of knee joint is 79°, and the extension limit is 0°. The schematic diagram of the extreme position of joint motion is shown in Fig. 2.

https://doi.org/10.1017/S0263574721001934 Published online by Cambridge University Press

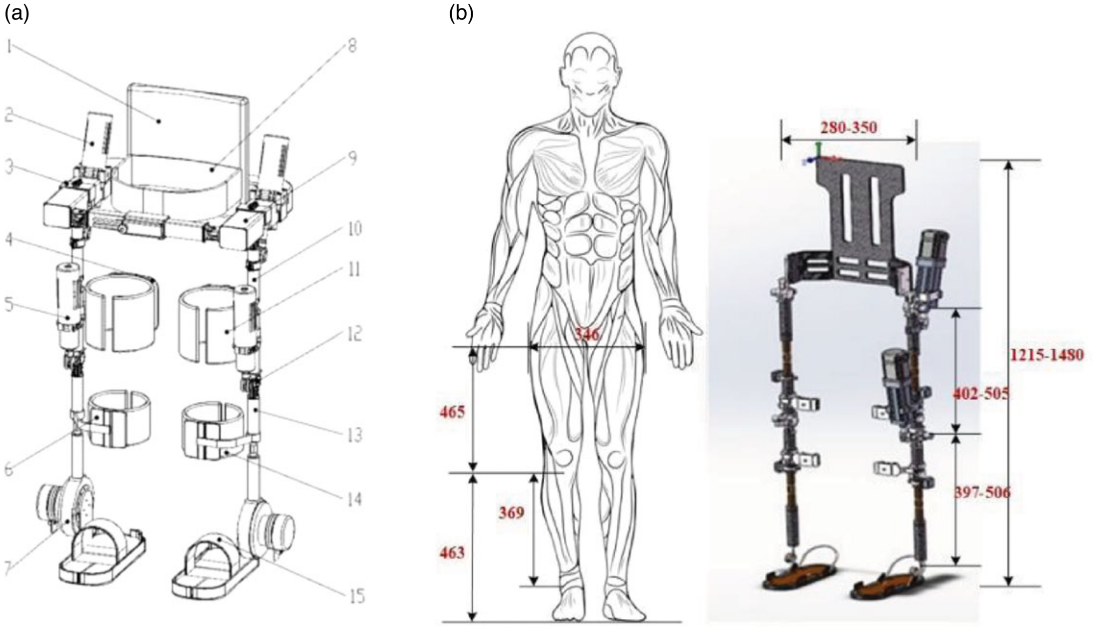

Figure 1. *(a) Overall structure diagram (1) Back support; (2) Driving source of hip joint; (3) Hip joint component; (4) Brace of thigh; (5) Driving source of knee joint;(6) Brace of calf; (7) Ankle joint component; (8) Flexible belt of waist;(9) Waist component;(10) Thigh component; (11) Flexible belt of thigh; (12) Knee joint component; (13) Calf component; (14) Flexible belt of calf; (15) Pedal; (b). Exoskeleton mechanical diagram.*

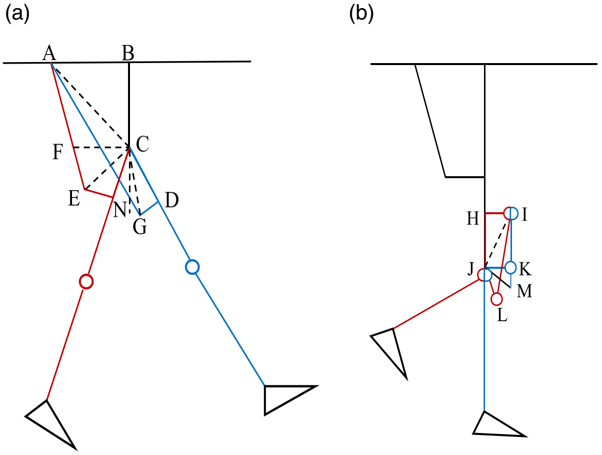

Figure 2. *The schematic diagram of the exoskeleton. (a) Motion limit position diagram of hip joint; (b) Motion limit position diagram of knee joint.*

Just as shown in Fig. 2(a), when the minimum flexion degree of hip joint is 40°:

$$AC = \sqrt{AB^2 + BC^2} \tag{1}$$

$$CE = \sqrt{CN^2 + NE^2} \tag{2}$$

$$\angle ACE = 180^\circ - 40^\circ - \arcsin\frac{AB}{AC} - \arcsin\frac{NE}{CE} \tag{3}$$

https://doi.org/10.1017/S0263574721001934 Published online by Cambridge University Press

$$AE = \sqrt{AC^2 + CE^2 - 2 \times AC \times CE \times \cos \angle ACE} \tag{4}$$

where AE indicates that the lead screw is fully retracted. $AB = 100$ mm, $BC = 57$ mm, $CD = CN = 90$ mm, $DG = NE = 50$ mm.

When the maximum extension degree of hip joint is 35°, $CE = CG$

$$\angle ACG = 360^\circ - 145^\circ - \arcsin\frac{AB}{AC} - \arcsin\frac{DG}{CG} \tag{5}$$

$$AG = \sqrt{AC^2 + CG^2 - 2 \times AC \times CE \times \cos \angle ACG} \tag{6}$$

where AG means that the lead screw extends a certain distance.

$$\Delta H = AG - AE \tag{7}$$

where $\Delta H = 70mm$ is stroke of ball screw of hip joint.

The stroke of the ball screw of the knee joint is shown in Fig. 2(b).

When the minimum extension degree of knee joint is 0°:

$$IJ = \sqrt{HI^2 + HJ^2} \tag{8}$$

$$\angle IJK = 90^\circ - \arcsin\frac{HI}{IJ} \tag{9}$$

$$IK = \sqrt{IJ^2 + KJ^2 - 2 \times IJ \times KJ \times \cos \angle IJK} \tag{10}$$

where IK indicates that the lead screw is fully retracted.

When the maximum flexion degree of knee joint is 79°:

$$\angle IJL = 180^\circ - 11^\circ\text{-}\arcsin\frac{HI}{IJ} \tag{11}$$

$$IL = \sqrt{IJ^2 + LJ^2 - 2 \times IJ \times LJ \times \cos \angle IJL} \tag{12}$$

where IL means that the lead screw extends a certain distance.

$$\Delta K = IL - IK \tag{13}$$

where $\Delta K = 55mm$ is stroke of ball screw of knee joint. Therefore, a lead screw with a stroke range of 100mm is selected to meet the requirements of the telescopic stroke of the ball screw

In order to ensure that the movement mode is closer to the human movement, the linear actuator of the servo motor and ball screw was selected. The telescopic travel of the ball screw can be calculated by the limit position of each joint. For the mechanical design of the exoskeleton, the thigh rod and calf rod are adjustable, from 402 to 505 mm and 397 mm to 506 mm, respectively. The motion range of the hip joint is −38°–50.18°. The motion range of the knee joint is 0°–95.53°. In order to ensure the safety, we make the joint activity thresholds of the mechanical design larger than the actual range of motion of the human body. Through the dynamic analysis of the lower extremity exoskeleton system, the driver of hip joint and knee joint adopted the servo motor with a rated power of 400 W. The length of the ball screw is 4 mm, the rated thrust is 2000 N, the maximum linear speedup is 100 mm/s, which is sufficient for assisting people in rehabilitation training. The step function is utilized to control the speed of the ball screw. The linear motion of the ball screw is transformed into the rotational motion of the hip joint and knee joint in the sagittal plane, which approximately fits the normal gait of a human. The speed of the ball screw is input to make the lower limb exoskeleton exercise in accordance with the approximated normal human gait. The driving force of the given motion is deduced to verify whether the thrust in the previous stage is enough. The stretch and contract of thigh, calf, and waist adopted stepless adjustable mechanism. To ensure the lightweight of the robot, the carbon-fiber material and aluminum alloys are adopted. The thigh, calf, and waist are made of carbon fiber instead of the aluminum alloys. Compared to the overall structure of all aluminum alloys, the weight is reduced by 25%. This exoskeleton leg fits

https://doi.org/10.1017/S0263574721001934 Published online by Cambridge University Press

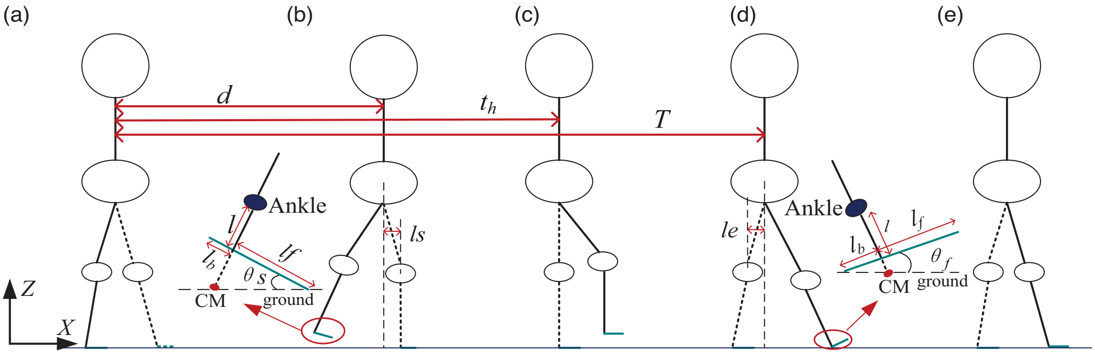

Figure 3. The state of five-point segmented gait planning method. (a). Dual support phase, (b). Initial stage of swing phase, (c). Middle stage of swing phase (d). Ending stage of swing phase (e). Second dual support phase.

patients from 1.50 to 1.90 m tall, which covers more than 99% of corresponding adults, with maximum body weight of 100 kg.

3. Gait trajectory generation

The lower extremity exoskeleton robot helps patients regain their ability to walk independently. Therefore, its walking mode would be planned according to walking stability, the standard step size, leg lift height, and gait cycle of ordinary people walking. Based on the cubic spline interpolation method, a five-point segmented gait planning method is proposed to the continuous walking trajectory.

Hypothesis 1: The upper body is always vertical to the horizontal ground.

Hypothesis 2: The *Z*-axis coordinate condition is always constant.

Hypothesis 3: The trajectory of the swinging leg is in the sagittal plane.

Hypothesis 4: During the swing phase, the hip and knee joints of the supporting leg will not experience internal/external rotation.

Hypothesis 5: When the swing leg is landing, there will be no slipping, stepping instability, etc.

The three-point (starting moment, swinging highest point moment, and landing moment) planning method was widely applied to gait planning. Although the technique is simple, too many changes in the intermediate process between the two states and excessive amplitude are easy to ignore. The five-point segmented planning method adds two conditions: the transition from dual support phase to swing phase and the transition from swing phase to second dual support phase, making the gait planning more elaborate, just as shown in Fig. 3.

3.1. Trajectory generation for ankle joint

In the swing phase, the step length of one step is set as s, and the angle between the sole of the foot and the ground is $\theta(t)$.

$$\theta(t)=\begin{cases} 0, t=kT \\ \theta_s, t=kT+d \\ 0, t=kT+t_h \\ \theta_f, t=(k+1)T \\ 0, t=(k+1)T+d \end{cases} \tag{14}$$

where k is positive integer, and $k \geq 0$. T is time of one gait cycle.

https://doi.org/10.1017/S0263574721001934 Published online by Cambridge University Press

During the swing phase, the posture of the swing leg is shown in Fig. 3(b)–(d). The center of mass (CM) of the sole was defined as the vertical point from the ankle joint to the foot. The length from the ankle joint to CM is represented by l. The length from the heel to CM is defined as l_b. The length from the toe to CM is defined as l_f.

The position of the ankle joint is

$$x(t)=\begin{cases} ks, t=kT \\ ks+l_f\,(1-\cos\theta_s)+l\sin\theta_s, t=kT+d \\ ks+L_h, t=kT+t_h \\ (k+l)s-l_b\,(1-\cos\theta_f)-l\sin\theta_f, t=(k+1)T \\ (k+1)s, t=(k+1)T+d \end{cases} \tag{15}$$

$$z(t)=\begin{cases} l, t=kT \\ l\cos\theta_s+l_f\sin\theta_s, t=kT+d \\ H, t=kT+t_h \\ l\cos\theta_f+l_b\sin\theta_f, t=(k+1)T \\ l, t=(k+1)T+d \end{cases} \tag{16}$$

where L_h is the distance from starting point at time t_h. H is the height of ankle joint.

Since at time kT and time $(k+1)T+d$, the supporting leg always touches the ground. Therefore, the constraint is:

$$\begin{cases} \theta'(kT)=0 \\ \theta'((k+1)T+d)=0 \end{cases} \tag{17}$$

$$\begin{cases} x'(kT)=0 \\ x'((k+1)T+d)=0 \end{cases} \tag{18}$$

$$\begin{cases} z'(kT)=0 \\ z'((k+1)T+d)=0 \end{cases} \tag{19}$$

3.2. Trajectory generation for hip joint

According to Hypothesis 2, the Z-axis coordinate of the hip joint is a constant, which is the sum of the length of the calf and the length of the thigh and the height of the ankle joint. When H is maximum, the distance traveled of the swinging leg is $s/2$.

$$x(t)=\begin{cases} ks+s/4, t=kT \\ ks+l_e, t=kT+d \\ ks+s/2, t=kT+t_h \\ ks+1_s, t=(k+1)T \\ ks+3s/4, t=(k+1)T+d \end{cases} \tag{20}$$

where $s/4<l_e<s/2, s/2<l_s<3s/4, l_s$, represents the distance between the position of hip joint of swinging leg and the supporting leg in the initial stage of swing phase. l_e represents the distance between the position of hip joint of swinging leg and the supporting leg in the ending stage of swing phase.

According to the initial speed and the final speed of the gait cycle, the constraint conditions can be obtained:

$$\begin{cases} x'(kt)=0 \\ x'((k+1)T+d)=0 \end{cases} \tag{21}$$

https://doi.org/10.1017/S0263574721001934 Published online by Cambridge University Press

4. Reinforcement learning with dynamic movement primitives

4.1. Dynamic movement primitives

DMP was flexible representation for motion primitives with good stability [20]. Any motion trajectory can be regarded as a combination of a second-order system and a nonlinear function. DMP can be described in the following form. According to the basic principle of dynamic motion primitives, in the process of trajectory learning, this method can obtain a motion trajectory that is the same as the teaching trajectory by providing teaching trajectory information, starting point value y_0, and the final point of the trajectory g.

$$\tau z_t' = \alpha_z(\beta_z(g - y_t) - z_t) + R_t \tag{22}$$

$$\tau y_t' = z_t \tag{23}$$

$$\tau x_t' = -\alpha_x x_t \tag{24}$$

$$R_t = h_t^T(\varsigma + \varepsilon_t) \tag{25}$$

where τ is a scaling factor related to time. y_t and y_t' are joint trajectory position and velocity. x_t and z_t represent the internal state. R_t is nonlinear function which determines the shape of the trajectory. The parameters α_x, α_z, β_z, and ς are scale factors. ε_t is interference term. h_t^T is a linear function approximator, which contains Gaussian kernels $\psi()$.

$$h_t = \frac{\sum_{k=1}^{N} \psi_k x_t}{\sum_{k=1}^{N} \psi_k}(g - y_0) \tag{26}$$

$$\psi_k = \exp\left(-\frac{1}{2\sigma_k^2}(x_t - c_k)^2\right) \tag{27}$$

where y_0 is the initial position of the trajectory. g is the final point of the trajectory. c_k and σ_k are the center and width of Gauss function. The parameter ς determines the shape of joint trajectory. The parameter ς will be adjusted by RL algorithm to update gait trajectories from y_0 to target g.

4.2. Stability criteria

Stable walking of the lower limb exoskeleton is the critical research content. Zero Moment Point (ZMP) is the point where the total moment of the human-exoskeleton system is 0 during walking, and its motion trajectory can be used as the basis for stability judgment. In the walking cycle, the smallest polygon formed by the contact points between the sole and the ground is called the supporting polygon. According to the definition of ZMP, if the human-exoskeleton system can walk stably, the ZMP trajectory must always fall within the supporting polygon. To prevent the occurrence of the ZMP trajectory from falling outside the edge of the supporting polygon, a certain distance is usually left at the boundary of the supporting polygon as a stability margin.

To realize the stable walking of the lower limb exoskeleton walking aid robot, this paper abstracts the stability problem as an objective function optimization problem. The objective function is constructed with the ZMP stability margin as a parameter. The shortest distance from ZMP to the edge of the stability margin is used to build the objective function. The coordinate origin is the point where the ankle joint of the supporting foot is projected on the ground. The distance between the toe of the supporting foot and the coordinate origin is D_{toe}. The distance between the heel of the supporting foot and the coordinate origin is D_{heel}. The center of stability margin of the supporting foot is $x_{fc} = (D_{toe}\text{-}D_{heel})/2$. If the exoskeleton system achieves stable walking, ZMP needs to meet $-D_{toe} < x_{zmp} < D_{heel}$. $f(xh)$ is an index function reflecting

https://doi.org/10.1017/S0263574721001934 Published online by Cambridge University Press

the deviation of the exoskeleton from the center of stability margin.

$$f(x_h) = (x_{zmp} - x_{fc}) \tag{28}$$

In this paper, a five-segment trajectory planning method is adopted, and the walking stability needs to be satisfied in each segment trajectory. In order to improve the similarity between the generation trajectory and the target trajectory, the objective function J was constructed. The smaller the J, the more stable the walking.

$$J = f(x_h) + \sum_{p=1}^{3} M_c g_p(x_p) + \sum_{q=1}^{3} \frac{M_c}{2} h_q(x_q) \tag{29}$$

$$g_p(x_p) = \begin{cases} 0, x'_h(t_p) > 0 \\ |x'_h(t_p)|, x'_h(t_p) < 0 \end{cases} \tag{30}$$

$$h_q(x_q) = \begin{cases} 0, -D_{toe} < x_{zmp} < D_{heel} \\ |x_{zmp}(t_q)|, x_{zmp}(t_q) < -D_{toe} \\ |x_{zmp}(t_q)|, x_{zmp}(t_q) > D_{heel} \end{cases} \tag{31}$$

where M_c is penalty factor, $g_p(x_p)$ and $hq(xq)$ are penalty function, $p, q = 1,2,3$, t_p and t_q are corresponding time consumption of penalty function.

4.3. Reinforcement learning with trajectory optimization

At present, the method of combining RL algorithm with other technologies has a good research prospect and has been widely used in many fields. Based on the stochastic Hamilton Jacobi Bellman method and path integral strategy learning method, a probabilistic RL method was proposed to solve the statistical reasoning problem from the continuous learning and training process of the sample data. The algorithm mainly uses sample data to train and learn for solving statistical reasoning problems and has the characteristics of simplicity, stability, and strong robustness. In this paper, the RL algorithm was used to update parameter ς, which could continuously update the shape of the trajectory to reach the target trajectory.

The cost function $S(\tau_k)$ of DMP was defined.

$$S(\tau_k) = \phi_{t_N} + \sum_{J=K}^{N-1} r_{t_j} + \sum_{J=k}^{N-1} \left\| \frac{y_{t_{j+1}} - y_{t_j}}{d_t} - R_{t_j} \right\|^2_{H_{t_j}^{-1}} + \frac{\lambda}{2} \sum_{j=k}^{N-1} \log\|H_{t_j}\| \tag{32}$$

$$\phi_{t_N} = B_N(y_t - g)^T(y_t - g) + R_N y_t'^T y_t' \tag{33}$$

$$r_t = \frac{1}{2} B_q y_t''^T y_t'' + \frac{1}{2} R(\varsigma + \varepsilon_t)^T(\varsigma + \varepsilon_t) \tag{34}$$

where ϕ_{tN} denotes the terminal cost at time t_N, B_N and R_N are applied for the adjustment of terminal cost, r_t represents the current cost at time t, B_q and R are used to change the current cost. H_t is a scalar, and $H_t = h_t^{(c)T} R^{-1} h_t^{(c)}$. $\frac{\lambda}{2} \sum_{j=i}^{N-1} \log \left|H_{tj}\right|$ is a certain variable, the values of all sampling paths are same, so it could be ignored.

Then, the expression of $S(\tau_i)$ is updated as:

$$S(\tau_k) = \phi_{t_N} + \sum_{N-1}^{j=k} r_{t_j} + \sum_{N-1}^{j=k} \left\| h_{t_j}^T(\varsigma_{t_j} + \varepsilon_{t_j}) \right\|^2_{H_{t_j}^{-1}} = \phi_{t_N} + \sum_{j=k}^{N-1} r_{t_j} + \sum_{j=k}^{N-1} \frac{1}{2}(\varsigma_{t_j} + \varepsilon_{t_j})^T h_{t_j} H_{t_j}^{-1} h_{t_j}^T (\varsigma_{t_j} + \varepsilon_{t_j})$$

$$= \phi_{t_N} + \sum_{j=k}^{N-1} r_{t_j} + \sum_{j=k}^{N-1} \frac{1}{2}(\varsigma_{t_j} + \varepsilon_{t_j})^T \frac{h_{t_j} h_{t_j}^T}{h_{t_j}^T R^{-1} h_{t_j}} (\varsigma_{t_j} + \varepsilon_{t_j}) \tag{35}$$

https://doi.org/10.1017/S0263574721001934 Published online by Cambridge University Press

According to the stochastic optimal control theory [21], the result of the path integral optimal problem for DMP can be expressed as:

$$\varsigma_{t_k} = \int P(\tau_k)\mu_L(\tau_k)d\tau_k \tag{36}$$

where $P(\tau_k) = \dfrac{\exp\left(-\frac{1}{\lambda}S(\tau_k)\right)}{\int \exp\left(-\frac{1}{\lambda}S(\tau_k)\right)d\tau_k}$ is probability variable, and the parameter λ was used to adjust the sensitivity of the exponential function. $\mu_L(\tau_k) = \dfrac{R^{-1}h_{t_k}h^T_{t_k}}{h_{t_k}R^{-1}h_{t_k}}\varepsilon_{t_k}$ is local control variable. And

$$S(\tau_k) = \phi_{tN} + \sum_{j=k}^{N-1} r_{tj} + \frac{1}{2}\sum_{j=k}^{N-1}(\varsigma_{t_k} + \varepsilon_{t_j})^T \times \left(\frac{R^{-1}h_{tj}h^T_{tj}}{h^T_{tj}R^{-1}h_{tj}}\right)^T R\left(\frac{R^{-1}h_{tj}h^T_{tj}}{h^T_{tj}R^{-1}h_{tj}}\right)(\varsigma_{t_k} + \varepsilon_{t_j}) \tag{37}$$

The stochastic optimal control problem is solved by Eq. (37) in the whole state space. The optimal control ς_{tk} can be obtained by an iterative update procedure. With an initial value ς, and a random parameter $\varsigma + \varepsilon_t$ is generated at each time step. With the consideration of Eq. (36), for the sake of the iterative updates, the update rule could be written as:

$$\varsigma^{new}_{tk} = \int P(\tau_k)\frac{R^{-1}h_{t_k}h^T_{t_k}(\varsigma + \varepsilon_{t_k})}{h^T_{t_k}R^{-1}h_{t_k}}d\tau_k = \int P(\tau_k)\frac{R^{-1}h_{t_k}\,h^T_{t_k}\varepsilon_{t_k}}{h^T_{t_k}\,R^{-1}h_{t_k}}d\tau_k + \frac{R^{-1}h_{t_k}\,h^T_{t_k}\varsigma_{t_k}}{h^T_{t_k}\,R^{-1}h_{t_k}} = \zeta\varsigma_{t_k} + \frac{R^{-1}h_{tj}\,h^T_{tj}}{h^T_{tj}\,R^{-1}h_{tj}}\varsigma \tag{38}$$

where $\zeta\varsigma_{t_k} = \int P(\tau_k)\dfrac{R^{-1}h_{t_k}h^T_{t_k}\varepsilon_{t_k}}{h^T_{t_k}R^{-1}h_{t_k}}d\tau_k$, $\varsigma^{new}_{t_k}$ is not time-independent. For each time step t_k, a new optimization parameter $\varsigma^{new}_{t_k}$ would be obtained. However, the time-independent parameter ς^{new} was needed, the parameter ς^{new}_{tk} was averaged over time t_k.

$$\varsigma^{new} = \frac{1}{N}\sum_{t=0}^{N-1}\varsigma^{new}_{t_k} = \frac{1}{N}\sum_{t=0}^{N-1}\zeta\varsigma_{t_k} + \frac{1}{N}\sum_{t=0}^{N-1}\frac{R^{-1}\,h_{t_k}\,h^T_{t_k}}{h^T_{t_k}\,R^{-1}\,h_{t_k}}\varsigma_{t_k} \tag{39}$$

The parameter ς^{new} includes two parts. The first term is the average value of $\zeta\varsigma_{t_k}$. It reflects the useful information obtained from the mixed noise information. The second term is the interference term, which causes the loss of parameter ζ. When the average value of $\zeta\varsigma_{t_k}$becomes 0, the convergence will be realized. Therefore, the second term does not need to further update. The second term should be eliminated. The parameter ς^{new} is updated to:

$$\varsigma^{new} = \frac{1}{N}\sum_{t=0}^{N-1}\zeta\varsigma_{t_k} + \frac{1}{N}\sum_{t=0}^{N-1}\varsigma_{t_k} = \frac{1}{N}\sum_{t=0}^{N-1}\zeta\varsigma_{t_k} + \varsigma \tag{40}$$

where ς^{new} is generated by the averaged value of $\zeta\varsigma_{t_k}$, and the current value of ς was added in each iteration.

The update of parameter ς is the most important part of the improved strategy algorithm, and it will affect the shape of the trajectory. The big difference between improved strategy and path integrals is using the combination of RL-based DMP and path integrals, where RL-based DMP can guarantee the stability of the system. The update rule of strategy improvement with path integrals is as follows. The parameters ς is mixed with noise interference; therefore, the noise interference ε_t was added to the parameter ς in DMP algorithm. While learning and updating the parameter ς, the updated result will lead to changes in the trajectory, and the corresponding cost function *S()* also changes. As the number of updates increases, the cost function *S()* gradually decreases and stabilizes to minimum value, and finally, the DMP system tends to be a noiseless system.

4.4. Segmented multiple-DMP trajectory learning method

When the DMP parameter values y_o and g are same, the learning trajectory is always a straight line, and the DMP method is invalid. However, the motion trajectory of the joint tends to be a sine function, and it is very likely that the values of y_0 and g are the same. Therefore, this paper proposed a segmented multiple-DMP trajectory learning method. Combined with five-point segmented gait planning method, the learned trajectory is divided into multiple monotonic trajectories. Because the starting value and target value of each segment of the learning trajectory after segmentation are not equal, the DMP method will not invalid. The specific steps of the segmented multiple-DMP trajectory learning method are as follows: The target trajectory was divided into multiple monotonous trajectories with five key moments as the dividing points. The target trajectory is divided into four segments. To learn each monotonous trajectory, set new starting value and target value, respectively. The target value of the previous segment is equal to the starting value of the following segment of the trajectory. The initial value of each segment of the target trajectory can be expressed as:

$$Y = [y_{KT}(k=0, y_{KT}=y_0), y_{KT+d}, y_{KT+t_h}, y_{(k+1)T}, y_{(k+1)T+d}] \tag{41}$$

The target value of the target trajectory can be expressed as:

$$G = [y_{KT+d}, y_{KT+t_h}, y_{(k+1)T}, y_{(k+1)T+d}, g] \tag{42}$$

The values of adjacent time points must be different, so the trajectory learning problem of $g = y_o$ is transformed into a multi-segment trajectory learning problem of $g \neq y_o$.

In the multi-DMP sequence, the target parameter g of a DMP is used as the starting parameter of the next DMP, and it can affect the cost of subsequent DMP. It is equivalent to using the cost of the current DMP and the cost of the remaining DMP in the sequence to optimize the shape parameter ς.

The cost function of multiple-DMP is defined as

$$S\left(\Pi_c\right) = \sum_{j=c}^{C} S(\tau_{t_0}^j) \tag{43}$$

where Π_c denotes the c section trajectory in the C sequences. $S(\tau_{t_0}^j)$ denotes the cost function of the j-th trajectory in C sequence at time t_0. $s(\Pi_c)$ denotes the total cost of the current trajectory and all subsequent sequential trajectories.

The parameter update rule of multi-DMP sequence is similar to that of single DMP, and the specific rules are as follows:

$$S(\tau_{t_{j,N}}) = \sum_{j=0}^{N} r_{t_j} + \varsigma_{t_j}^T R \varsigma_{t_j} \tag{44}$$

$$S\left(\Pi_{c,k}\right) = \sum_{i=c}^{C} S(\tau_{t_0,k}^j) \tag{45}$$

$$P\left(\Pi_{c,k}\right) = \frac{\exp\left(\left(-\frac{1}{\lambda} S\left(\Pi_{c,k}\right)\right)\right)}{\sum_{l=1}^{K} \exp\left(\left(-\frac{1}{\lambda} S\left(\Pi_{c,l}\right)\right)\right)} \tag{46}$$

$$\zeta\varsigma_c = \sum_{k=1}^{K} P\left(\Pi_{c,k}\right) \varepsilon_k^{\varsigma c} \tag{47}$$

$$\varsigma_c^{new} = \varsigma_c + \zeta\varsigma_c \tag{48}$$

where Eq. (44) is the cost function of a certain trajectory; Eq. (45) is the calculation of the cost of all sequences.

https://doi.org/10.1017/S0263574721001934 Published online by Cambridge University Press

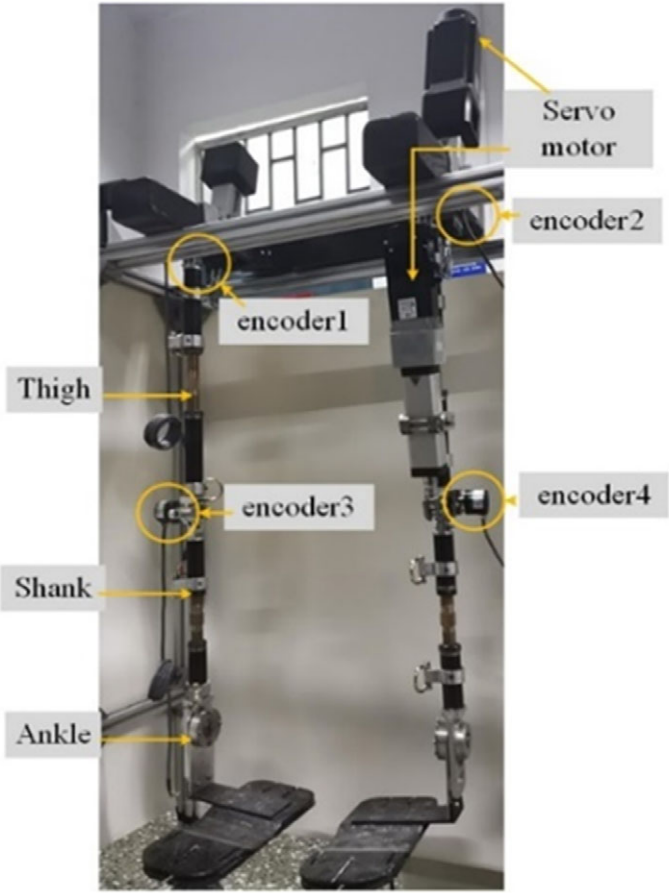

Figure 4. *Prototype of the proposed exoskeleton robot.*

5. Experimental verification

5.1. Experimental platform

This paper describes a novel gait trajectory planning method based on multiple-DMP and RL algorithm for lower exoskeleton robot. In Fig. 4, the exoskeleton robot system consists of three levels of architecture, including an upper-level controller, a lower limb exoskeleton robot, and a low-level controller. The upper-level control system consists of the data processor Raspberry Pi, which is used to process data collected from encoders (AD36/1217AF. ORBVB, Hengstler, Germany). The joint movement data of the hip joint were acquired by encoder 1 encoder 2, encoder 3, and encoder 4 together. The low-level controller controls the robot to perform trajectory. The communication mode between the upper-level and the low-level controller is parallel port communication. The low-level controller is used to control the motors through the Controller Area Network (CAN) Fieldbus. The experiment was conducted in Tianjin Key Laboratory for Integrated Design & Online Monitor Center. Two healthy male subjects (subject A and subject B) were recruited for the experiment, with averaged age 24 years old, averaged height 174 cm, averaged weight 71.73 kg. The subjects were instructed to perform normal walking with exoskeleton on the ground. For safety reasons, there are two levels of safety measures taken. Firstly, if the angle of the exoskeleton is greater than the allowable value (The motion range of the hip joint is $-38°$–$50.18°$. The motion range of the knee joint is $0°$–$95.53°$), the exoskeleton will be forced to stop moving by software. Besides, the subjects could cut off the power through the emergency switch button if they need to do it at any time, just as shown in Fig. 5.

https://doi.org/10.1017/S0263574721001934 Published online by Cambridge University Press

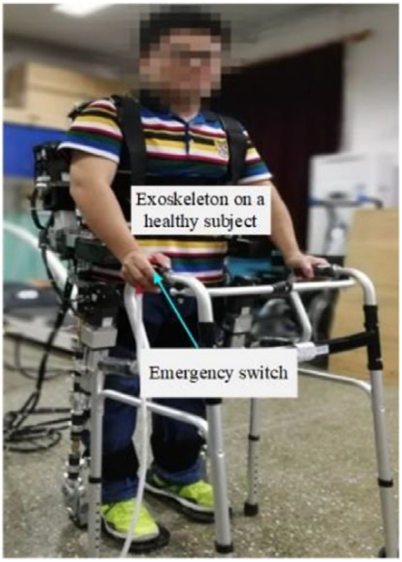

Figure 5. *Human experiment.*

5.2. Parameters setting

The time of dual support phase d is 0.2 s. The one-step length s is 0.6 m. The maximum height of ankle joint H is 0.25 m, and the elapsed time t_h is 0.5 s. The parameters l_b, l, $l_{s,}$ and l_e are set as 0.06, 0.1, 0.4, and 0.2 m, respectively. The length of calf is 0.3 m. The length of thigh is 0.5 m. The height of hip joint is 0.7 m. The velocity of horizontal hip of subject A and subject B is 0.8 m/s and 0.67 m/s, respectively. The parameters of DMP equation are $\alpha_z = 20$, $\beta_z = 20$, $\tau = 0.2$. The parameters of cost function are $BN = 10^{-5}$, $R = 10^{-5}$, $B_N = 10$, $R_N = 25$. The parameters of stability criteria are $D_{toe} = 0.15$ m, $D_{heel} =$ 0.1 m, $M_c = 300$.

5.3. Experimental results

In the experiment, the trajectory learning of all joints of the lower limb exoskeleton is updated at the same time. The number of updates is set to 70. After each update, DMP generates the required joint trajectory. Then, the actual joint trajectory is obtained. The data from the actual joint trajectory are used as the value of the RL algorithm to calculate the corresponding cost, and the shape and target parameters are updated. After the shape and target parameters are updated, the multiple-DMP algorithm is used to generate a new desired joint trajectory for the lower limb exoskeleton.

The experimental results are presented in Figs. 6–8. Figures 6 and 7 show the actual learning trajectories of the two joints in the lower exoskeleton. In order to show the effect of learning clearly, six groups of data are obtained by sampling data every ten times. Each joint trajectory is composed of four sequential segments. Due to the environmental disturbance, the learning trajectory at the initial time has a larger state error. Although the subjects' movement and speed were inconsistent, the stable convergence of the two tests is achieved through continuous learning. The experimental results show that the reinforcement learning algorithm can well suppress uncertainty and interference. After 30 iterations, with the increase of the number of updates, the amplitude is gradually stable and convergent, which verifies the learning performance of the reinforcement learning algorithm, and shows that the joint trajectory is convergent. The change in the total cost value of all trajectories is shown in Fig. 8. The amplitude tends to stabilize and converge as the number of updates increases, which verifies the validity and superiority of the proposed method. The video capture of the human experiment is shown in Fig. 9. The time of a complete

https://doi.org/10.1017/S0263574721001934 Published online by Cambridge University Press

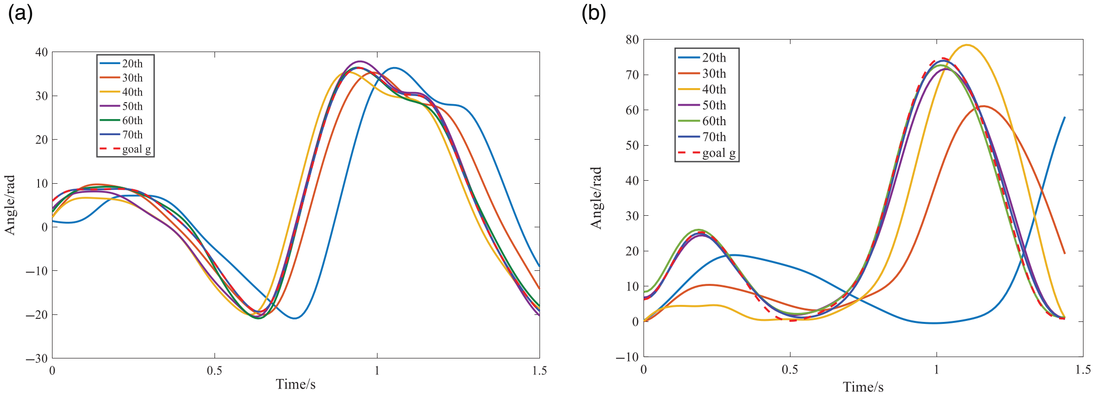

Figure 6. *Actual learning trajectories of joints of subject A. (a) Trajectory of hip joint. (b) Trajectory of knee joint.*

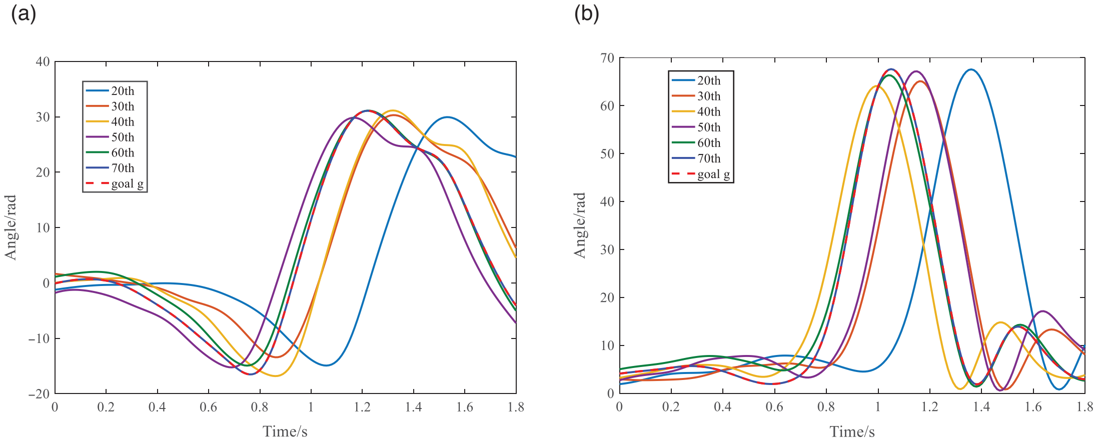

Figure 7. *Actual learning trajectories of joints of subject B. (a) Trajectory of hip joint. (b) Trajectory of knee joint.*

gait cycle of subject A is 1.5 s. The experiment demonstrates that the exoskeleton robot can effectively assist the human body to realize stable walking.

5.4. Stability analysis

Based on the D'Alembert's principle, when the exoskeleton and human body were regarded as the multi links systems, the ZMPs could be calculated by Eqs. (49) and (50) [32]. So the ZMP positions of the exoskeleton and human body in the coordinate system could be determined when walking.

$$x_{zmp} = \frac{\sum_i m_i(z_i'' + G_g)x_i - \sum_i m_i x_i'' z_i}{\sum_i m_i(z_i'' + G_g)} \tag{49}$$

$$y_{zmp} = \frac{\sum_i m_i(z_i'' + G_g)y_i - \sum_i m_i y_i'' z_i}{\sum_i m_i(z_i'' + G_g)} \tag{50}$$

where G_g is gravitational acceleration. (x_i, y_i, z_i) locates the center of mass of link-i, and m_i represents the mass of link i.

https://doi.org/10.1017/S0263574721001934 Published online by Cambridge University Press

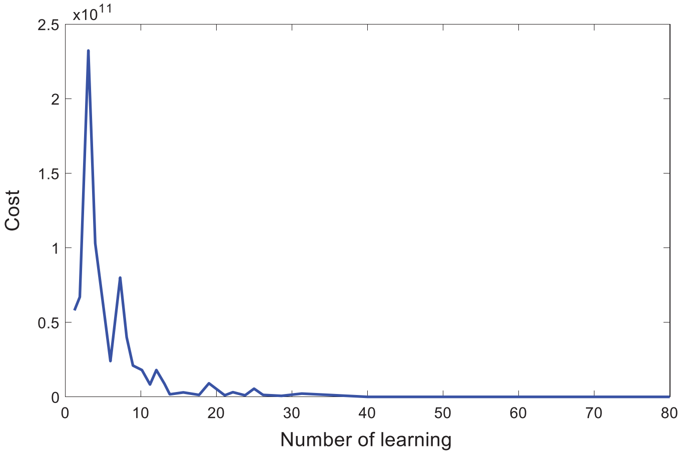

Figure 8. *Cost during reinforcement learning of subject A.*

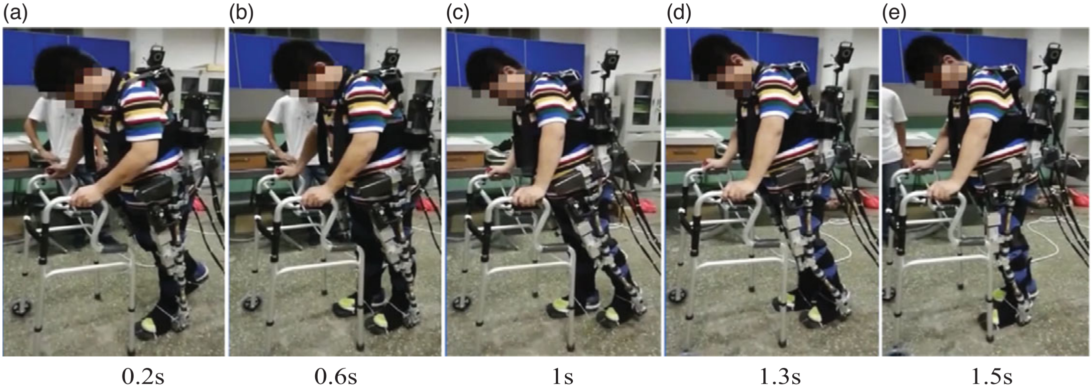

Figure 9. *The video snapshot of subject A's experiment.*

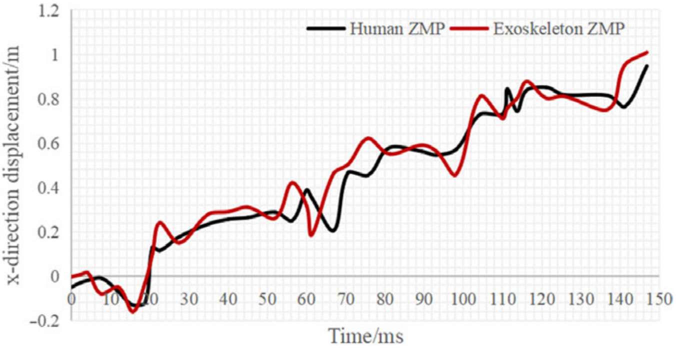

Figure 10. *The position trajectories of body ZMP and exoskeleton ZMP.*

Only the Sagittal plane was considered, and the position trajectories of body ZMP and exoskeleton ZMP are shown in Fig. 10. From the graph, we could see the ZMP of exoskeleton could track the wear's ZMP well. And there is no greater ZMP gap caused by the interference. The whole system could run stably and effectively.

https://doi.org/10.1017/S0263574721001934 Published online by Cambridge University Press

6. Conclusion

This paper proposes a new method that can help the human body walk stably with the lower limb exoskeleton robot. The mechanical structure and robot system of the 6-DOFs exoskeleton robot is introduced. A new five-point walking trajectory planning method for the lower limb exoskeleton robot is proposed. The novel stability criteria of human-exoskeleton system are based on ZMP. And we could see the ZMP of exoskeleton could track the wear's ZMP well from experimental results. Besides, the multiple DMP strategy based on reinforcement learning is used to transform the trajectory of task space into the trajectory of angle space, which reduces the influence of external systems and various disturbances. It can be seen from the experiment that with the participation of multiple-DMP algorithm based on RL algorithm, the trajectory of the exoskeleton robot could be continuously adjusted to track the target trajectory smoothly. The current research content is mainly for simple locomotion scenes. Future work will consider the concept of the adaptive training strategy to strengthen the universality of the exoskeleton robot.

Consent to Publish. The work described has not been submitted elsewhere for publication, in whole or in part, and all the authors listed have approved the manuscript that is enclosed.

Authors Contributions. All authors have made substantial contribution. The first author Peng Zhang designed the experiment scheme, analyzed the experimental results, and wrote the paper. Corresponding author Junxia Zhang modified the paper.

Funding. This work was supported in part by the Hundred Cities and Hundreds Parks Project-Key Nutrition and Health Technology and Intelligent Manufacturing(21ZYQCSY00050).

References

[1] H. Kazerooni and R. Steger, "The Berkeley lower extremity exoskeleton," *J. Dyn. Syst. Meas. Control* **128**(1), 14–25 (2006).

[2] E. Guizzo and H. Goldstein, "The rise of the body bots," *IEEE Spectrum* **42**(10), 42 (2005).

[3] G. T. Huang, "Wearable robots," *Technol. Rev.* **28**(5), 70–73 (2004).

[4] C. J. Walsh, K. Pasch and H. Herr, "An Autonomous, Underactuated Exoskeleton for Load-Carrying Augmentation," *2006 IEEE/RSJ International Conference on Intelligent Robots and Systems* (IEEE, 2006) pp. 1410–1415.

[5] C. J. Walsh, "Biomimetic design of an under-actuated leg exoskeleton for load-carrying augmentation," *Massachusetts Inst of Tech Cambridge Media Lab* (2006).

[6] C. J. Walsh, K. Endo and H. Herr, "A quasi-passive leg exoskeleton for load-carrying augmentation," *Int. J. Humanoid Rob.* **4**(3), 487–506 (2007).

[7] Y. Sankai, "HAL: Hybrid assistive limb based on cybernics," *Rob. Res.* **1**(66), 25–34 (2010).

[8] A. Esquenazi, M. Talaty, A. Packel and M. Saulino, "The re walk powered exoskeleton to restore ambulatory function to individuals with thoracic-level motor-complete spinal cord injury," *Am. J. Phys. Med. Rehabil.* **91**(11), 911–921 (2012).

[9] B. H. Hu, N. E. Krausz and L. J. Hargrove, "A Novel Method for Bilateral Gait Segmentation Using a Single Thigh-Mounted Depth Sensor and IMU," *IEEE International Conference on Biomedical Robotics and Biomechanics, Enschede* (2018) pp. 807–812.

[10] K. Chao and P. Hur, "A Step Towards Generating Human-Like Walking Gait Via Trajectory Optimization Through Contact for a Bipedal Robot with One-Sided Springs on Toes," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, *Vancouver, BC* (2017) pp. 4848–4853.

[11] S. Faraji and A. J. Ijspeert, "Scalable Closed-Form Trajectories for Periodic and Non-Periodic Human-Like Walking," *International Conference on Robotics and Automation (ICRA)*, *Montreal, QC, Canada* (2019) pp. 5295–5301.

[12] F. Wang, Y. Wang, S. Wen and S. Zhao, "Nao Humanoid Robot Gait Planning Based on the Linear Inverted Pendulum," *Chinese Control and Decision Conference (CCDC)*, *Taiyuan* (2012) pp. 986–990.

[13] M. Kasaei, N. Lau and A. Pereira, "A Fast and Stable Omnidirectional Walking Engine for the Nao Humanoid Robot," Robot World Cup XXIII (2019) pp. 99–111.

[14] B. Sebastian, H. Ren and P. Ben-Tzvi, "Neural Network Based Heterogeneous Sensor Fusion for Robot Motion Planning," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), Macau, China* (2019) pp. 2899–2904.

[15] J. Demby, Y. Gao and G. N. De Souza, "A Study on Solving the Inverse Kinematics of Serial Robots using Artificial Neural Network and Fuzzy Neural Network," *IEEE International Conference on Fuzzy Systems (FUZZ-IEEE), New Orleans, LA, USA* (2019) pp. 1–6.

[16] H. Wang, T. Lu, B. Niu, H. Yan, X. Wang, J. Chen and Y. Li, "Research on Fuzzy PID Control Algorithm for Lower Limb Rehabilitation Robot," *IEEE 4th Information Technology and Mechatronics Engineering Conference, Chongqing, China* (2018) pp. 956–960.

[17] Y. Wen and L. Huiyi. "Gait optimization of humanoid robot based on deep Q-network," *Comput. Modernizat.* **56**(4), 47–58 (2019).

[18] A. A. Saputra, J. Botzheim, I. A. Sulistijono and N. Kubota, "Biologically inspired control system for 3D locomotion of a humanoid biped robot," *IEEE Trans. Syst. Man Cybern. Syst.* **7**(46), 898–911 (2016).

[19] A. Ijspeert, J. Nakanishi and S. Schaal. "Movement Imitation with Nonlinear Dynamical Systems in Humanoid Robots," IEEE International Conference on Robotics and Automation (ICRA2002) (2002) pp. 398–403.

[20] C. L. Bottasso, D. Leonello and B. Savini, "Path planning for autonomous vehicles by trajectory smoothing using motion primitives," *IEEE Trans.* Control Syst. *Technol*. **16**(6), 1152–1168 (2008).

[21] F. Stulp, E. A. Theodorou and S. Schaal, "Reinforcement learning with sequences of motion primitives for robust manipulation," *IEEE Trans. Robot*. **28**(6), 1360–1370 (2012).

[22] E. Theodorou, J. Buchli and S. Schaal, "A generalized path integral control approach to reinforcement learning," *J. Mach. Learn. Res.* **11**(11), 3137–3181 (2010).

[23] R. Nian, J. Liu and B. Huang, "A review on reinforcement learning: Introduction and applications in industrial process control", *Comput. Chem. Eng.* **139**, 1–30 (2020).

[24] J. De Jesús Rubio, "Discrete time control based in neural networks for pendulums," *Appl. Soft Comput.* **68**(11), 821–832 (2018).

[25] X. Gao, B. Sun, X. Hu and K. Zhu, "Echo state network for extended state observer and sliding mode control of vehicle drive motor with unknown hysteresis nonlinearity," *Math. Probl. Eng.* **31**(13), 1–13 (2020).

[26] J. Zhao, "Neural networks-based optimal tracking control for nonzero-sum games of multi-player continuous-time nonlinear systems via reinforcement learning," *Neurocomputing* **412**(13), 167–176 (2020).

[27] M. Naeem, S. T. H. Rizvi and A. Coronato, "A gentle introduction to reinforcement learning and its application in different fields", *IEEE Access* **8**(11), 209320–209344 (2020).

[28] E. Theodorou, J. Buchli and S. Schaal, "Reinforcement Learning of Motor Skills in High Dimensions: A Path Integral Approach," *2010 IEEE International Conference on Robotics and Automation* (2010) pp. 2397–2403.

[29] B. Luo, D. Liu, T. Huang and D. Wang, "Model-free optimal tracking control via critic-only Q-learning," *IEEE Trans. Neural Networks Learn. Syst.* **27**(10), 2134–2144 (2016).

[30] J. Peng and R. J. Williams, "Incremental multi-step Q-learning," *Mach. Learn.* **22**(7), 283–290 (1996).

[31] J. A. Lázaro-Camí, "The stochastic Hamilton-Jacobi equation," *J. Geom. Mech.* **1**(3), 295–315 (2008).

[32] N. Aphiratsakun, K. Chairungsarpsook and M. Parnichkun, "ZMP Based Gait Generation of AIT's Leg Exoskeleton," 2010 *The 2nd International Conference on Computer and Automation Engineering (ICCAE)*, vol. 5 (2010) pp. 886–890.

Cite this article: P. Zhang and J. Zhang (2022). "Motion generation for walking exoskeleton robot using multiple dynamic movement primitives sequences combined with reinforcement learning", Robotica **40**, 2732–2747. https://doi.org/10.1017/S0263574721001934