IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received February 8, 2020, accepted February 18, 2020, date of publication February 24, 2020, date of current version March 27, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2976098

Workspace Trajectory Generation Method for Humanoid Adaptive Walking With Dynamic Motion Primitives

CHENGJU LIU[1], WANDONG GENG[1], MING LIU[2], (Senior Member, IEEE), AND QIJUN CHEN[1], (Senior Member, IEEE)

[1]Department of Control Science and Engineering, Tongji University, Shanghai 201804, China
[2]Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong 999077

Corresponding author: Qijun Chen (qjchen@tongji.edu.cn)

This work was supported in part by the National Natural Science Foundation of China (Grant No. 61673300, U1713211), and Basic Research Project of Shanghai Science and Technology Commission (Grant No. 18DZ1200804).

ABSTRACT To properly function in real-world environments, a humanoid robot must be able to adapt its walking gait to new situations. In this paper, an adaptive bipedal walking control method that uses sensory feedback to modulate dynamic movement primitive (DMP) parameters is presented. This work addresses the challenge of adaptive locomotion by implementing DMPs in the workspace of a humanoid robot. This workspace formulation allows new movements to be created such that the DMP parameters, including the stride, height of the hip joint, foot clearance and forward velocity, are directly related to the walking pattern. One set of DMPs is applied to generate the foot trajectory, and a second set is used to generate the CoM (centre of mass) trajectory in an online fashion. Sensory feedback information is utilized to modify the generated CoM and foot trajectories to improve the walking quality. Furthermore, a staged evolutionary algorithm (EA) is designed to optimize the parameters of the control system to enhance the walking performance. The presented control strategy is demonstrated through simulations and real experiments that focus on the adaptation of the robot's walking pattern over sloped terrain.

INDEX TERMS Humanoid robot, adaptive walking, dynamic movement primitives (DMPs), workspace trajectory generation.

I. INTRODUCTION

Biped locomotion is complex to analyze in general. Most of the models in the literature for biped locomotion utilize hybrid dynamics and various control strategies have been proposed [1]–[8]. ZMP (zero moment point) -based walking control methods are usually applied on bipedal robots [5]–[8]. The ZMP criterion was proposed by Vukobratovic *et al.* [9] and Vukobratovic and Borovac [10]. The main idea of the ZMP criterion is that if the ZMP of a bipedal robot lies within the support polygon formed between the foot and the ground, then stability is guaranteed. Currently, strategies based on trajectory planning are suitable for allowing a robot to walk stably by tracking pre-designed trajectories. However, such fixed pre-designed trajectories may cause the walking process to fail when the walking conditions vary. A bipedal robot walking under different conditions must be able to adapt to terrain variations by adjusting its gait trajectories. Park *et al.* [11] proposed a walking trajectory generation strategy based on combining sinusoidal functions and 3rd-order polynomial functions to achieve free biped walking gait. Seven *et al.* [12] employed a gait synthesis technique based on the ZMP in which the body pitch reference for the foot sole plane was modified online to adapt to different slope environments by using a fuzzy logic system. Omnidirectional walking on sloped terrain was studied by Yu *et al.* [13]. The foot trajectories were determined by the stride length, walking speed and direction, and the robot's CoM trajectories in both the coronal and sagittal planes were calculated using a linear inverted pendulum model (LIPM).

DMPs are widely applied on fault diagnosis and fault tolerant control [14]–[16], and the attribute of DMPs is attractive for designing smooth kinematic control algorithms that can robustly imitate demonstrated movements [17]–[19].

The associate editor coordinating the review of this manuscript and approving it for publication was Shihong Ding.

This work is licensed under a Creative Commons Attribution 4.0 License. For more information, see https://creativecommons.org/licenses/by/4.0/

The fundamental theory of DMPs is based on the utilization of an analytical dynamic system that is well understood and has convenient stability properties; this system is regulated by varying the nonlinear part to achieve a certain ideal limit cycle or point attractor. By using the learnable forcing function, the simple attractor can be transformed, as originally presented by Ijspeert *et al.* [20], [21]. Since the original proposal, many mathematical variants have been presented for the realization of periodic and discrete movements. DMPs provide a simple and adaptable method for motor skill generation and used in various tasks, like drumming [22]–[25]. Schaal [22] described the implementation of a system of DMPs on a complex anthropomorphic robot. Umlauft *et al.* [25] proposed a control strategy for synchronization and cooperative manipulation based on cooperative DMPs. Many studies involving DMPs have studied methods for generalizing and adjusting learned behaviours to adapt to novel situations. For example, given a learned grasping behaviour, obstacle avoidance [26], [27] can be realized by adding a forcing term that induces movement away from obstacles. Frequency adaptive modulation has been investigated as a form of temporal coupling to realize desirable drumming behaviours [28]. Reinforcement learning for DMPs has been used to guide policy searches and enhance learned behaviours [29]–[31]. Beyond the modelling of motion in Cartesian space, Ude *et al*. proposed a task-specific generation method for discrete and periodic DMPs to incorporate end-effector orientation [32].

DMPs are a useful tool for designing trajectories based on presented examples. This method is currently regarded as a representative example of a compact strategy that is suitable for robot learning. DMPs are typically learned in the joint space of robots, as reported by Nakanishi *et al.* [33] and Morimoto *et al.* [34]. We call this approach the ''joint-space DMP-based method''. However, investigating the adaptation and generalization of learned primitives through control parameter modulation becomes challenging when the demonstrated trajectories are available only in the joint space.

A bipedal robot walking under variable walking conditions, e.g., in the presence of obstacles, must adapt to its walking terrain by means of adjustable workspace trajectories. Unlike in the abovementioned works, the DMPs used for this purpose must be learned in the task space; we call this approach the ''workspace DMP-based method''. This idea of using DMPs in the task space, which can be easily related to the task parameters, was first presented by Pastor *et al.* [35] and Rosado *et al.* [36] in the context of robot manipulation and bipedal walking. Subsequently, various task-space methods have been studied to improve the robustness and stability of bipedal motion by achieving objectives such as push recovery and disturbance resistance. Luo *et al*. presented bio-inspired push recovery strategies based on DMPs [37]. Using DMPs, Bockmann *et al*. realized adaptive kick motions for the NAO robot [38]. In these strategies, DMPs are utilized as trajectory representations learned in the task space from a single demonstration. Once the DMPs have been learned, new movements are produced by directly revising the DMP parameters. The DMP parameters resulting from the related formulas are directly associated with task variables such as the stride length, foot clearance and forward velocity. For instance, the frequency parameter is utilized to slow down or speed up locomotion, and the amplitude parameters related to the y and z directions are employed to regulate the stride length and hip height of the supporting leg (or swing leg), respectively.

Our work considers how to incorporate sensory information to improve the adaptability of workspace DMP-based bipedal walking. In this paper, DMPs are used to plan the workspace and CoM trajectories of a humanoid robot. Two sets of DMP units are used to create a trajectory generator that can be modulated online by training the example trajectory. The robot's own attitude information is measured by sensors, and these measurements form the feedback input to the trajectory generator. The CoM and foot trajectories can be adjusted in real time. Notably, changes in the walking pattern are realized by simply modifying a few parameters of the trajectory generator based on the sensory information. These adjustments do not require prior information concerning the walking conditions, nor do they rely on range sensors for surface topography measurements. Both simulations and real experiments are designed to verify the effectiveness of the presented control system.

The other sections of the paper are arranged as follows: Section II presents the structure and properties of the proposed control framework. The design methods for the trajectory generators and the feedback path are introduced in detail, followed by the evolution of the control parameters. Section III presents computational simulations conducted to evaluate the NAO robot's adaptive walking behaviour. In Section IV, this paper is concluded, and future research directions are discussed.

II. THE PROPOSED WORKSPACE DMP-BASED CONTROL SYSTEM

This section demonstrates the structure and implementation of the presented adaptive bipedal walking control method. The control system architecture as shown in Fig.1, consists of a CoM trajectory generator, a workspace trajectory generator and a motion engine.

The CoM generator is trained using a DMP model by studying a demonstrated trajectory of a humanoid robot. Similar to the training of the CoM trajectory generator, two DMPs in the same canonical system are trained and employed as foot trajectory generators in the x and z directions; their final one-step positions correspond to the length of the step and the height of the leg lift, respectively. Thus, the gait of the humanoid robot can be adjusted by modulating the DMP parameters. The regulation of the swing foot trajectory should be combined with that of the CoM trajectory to best enhance the adaptability and stability of the walking motion. The trajectory generation methods are presented in the following sections.

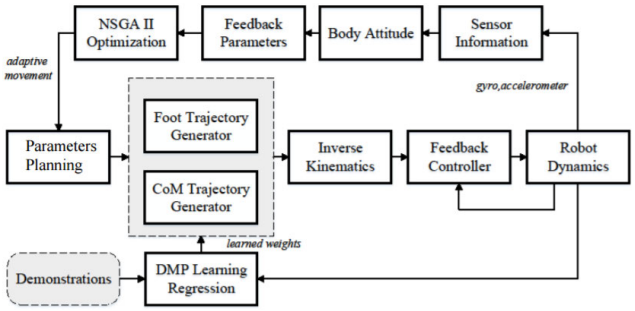


FIGURE 1. The global control architecture.

A. DMP-BASED TRAJECTORY FORMULATION

1) DMP MODEL

The dynamical system is described by a linear 2nd-order differential equation, in which convergence to the goal g is achieved by adding a nonlinear forcing term f that represents the actual shape of the encoded trajectory. In our work, DMPs are employed to represent bipedal walking movements. DMPs are formulated for each x, y and z coordinate of the two feet in the workspace: the origin of the reference coordinate system is located at the hip section of the robot. This approach yields a total of four DMPs, whose outputs are converted into the desired joint trajectories through an inverse kinematics algorithm and are utilized as the reference input to the low-level feedback controller. The differential equations utilized to describe each single DMP are presented as follows.

The transformation system is defined by two 1st-order differential equations:

$$\tau \dot{\upsilon} = \alpha_\upsilon \left(\beta_\upsilon \left(g - y + feed\right) - \upsilon\right) + f \tag{1}$$

$$\tau \dot{y} = \upsilon \tag{2}$$

The canonical system describes the phase of the movement:

$$\tau \dot{u} = -\alpha_u u \tag{3}$$

The forcing term f is as follows:

$$f\left(t\right) = \frac{\sum_{i=1}^{N} \psi_i(t)\omega_i}{\sum_{i=1}^{N} \psi_i(t)} u\left(g - y_0\right) \tag{4}$$

$$\psi_i(u) = \exp\left(-\frac{1}{2\sigma_i^2}\left(u - c_i\right)^2\right) \tag{5}$$

where α_υ, β_υ and α_u are time constants; y, v and $\dot{v}$ denote the position, velocity, and acceleration, respectively; τ is a temporal scaling constant; y_0 is the initial state; g is a known goal; and *feed* is the feedback term that receives the feedback information. Here, f is a nonlinear forcing function that acts as the forcing term to represent the nonlinearity of the model. When $f = 0$, this equation acts as a global stabilization system in which the target point g serves as an attractor. The system can be critically damped by choosing appropriate values for α_υ and β_υ ($\alpha_\upsilon = \beta_\upsilon/4$) such that y will monotonically converge towards g. The transfer system and the canonical system have the same time constant τ. The phase variable u acts to normalize the system's convergence time over the range from 1 to 0. To avoid any time dependence of the entire nonlinear term, i.e., to ensure that the system is invariant with respect to time, the entire movement time depends only on τ. The ω_i represent adjustable weights, the $\psi_i(u)$ are exponential basis functions, and the σ_i and c_i are constants that determine the widths and centres, respectively, of the basis functions.

A demonstration trajectory $y_{demo}(t)$ should be provided when using DMPs for trajectory learning. In this work, the demonstration trajectories $y_{demo}(t)$ for the movement primitive mode are generated by a real humanoid robot. To encode an ideal demonstration trajectory $y_{demo}(t)$ as a DMP, the weight vector must be learned by using statistical learning techniques in which locally weighted regression (LWR) is adopted. Therefore, local models are used to attempt to fit the data only in the region around the position of the query by utilizing distance-weighted regression rather than using an individual global model to fit all the training data. The motivation for investigating locally weighted techniques originates from their suitability for online robot learning by virtue of their fast incremental learning properties. LWR is used to perform training and obtain the parameters of the Gaussian kernel functions $\psi_i(u)$ and the parameters ω_i. For different training requirements, only the kernel function bandwidth is adjusted. When the trajectory to be learned is relatively complex, the number of kernel functions can be increased. After training is complete, the trajectory generation process depends on the DMPs.

2) PARAMETERS MODULATION

The DMP model contains certain parameters that can influence the output behaviour in accordance with different control demands. This section discusses how movements are adjusted to adapt to new situations by changing the DMP parameters. The final goal position and duration of a movement are revised by scaling parameters g and τ.

Fig. 2 presents the results of modulating the DMP parameters. All feedback information is set to 0. The black solid line represents the demonstration trajectory, which has a duration of 1.5 s and a final goal position of 0. The black dashed line, which represents the learned trajectory with no modulation, nearly coincides with the demonstration trajectory. The red solid line displays the result of changing the duration to 1 s by modulating the parameter τ, and the blue dashed line represents the result of changing the final goal position to 1 m by modulating the parameter g. The main observation is that modulating the parameters results in smooth trajectory variations that enable adaptation to different control demands.

3) CoM TRAJECTORY GENERATOR

This section describes the training of a CoM trajectory generator using a DMP model based on a demonstration trajectory of a humanoid robot. The CoMz trajectory can be mapped to the vertical-direction trajectory of the swing foot. Thus, two DMPs are trained to serve as the CoMx and CoMy

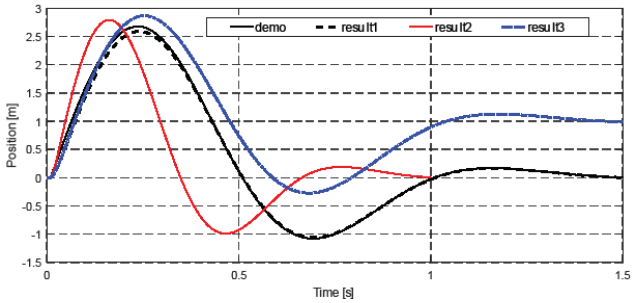


FIGURE 2. The parameters modulation of the DMP model.

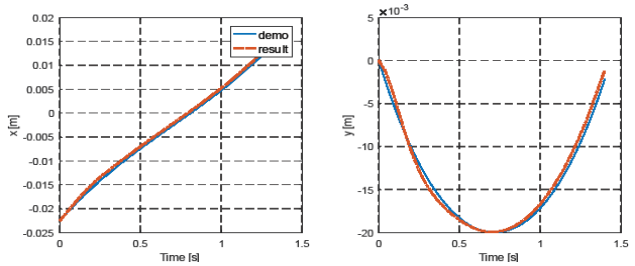


FIGURE 3. CoMx and CoMy trajectories for four steps.

trajectory generators. These two DMPs share the same canonical system but each have their own transformation systems to maintain suitable synchronization. Fig. 3 shows the results of the trained CoM trajectory generator and presents the CoM trajectories for four steps by the robot. The blue solid lines represent the demonstration trajectory, and the red dashed lines represent the output of the CoM trajectory generator.

4) FOOT TRAJECTORY GENERATOR

The robot's walking speed, leg lift height and step span must be flexibly adjustable to ensure adaptive walking. When the robot is walking, the trajectory of the swing foot is set to be constant in the y direction; thus, DMPs for the trajectories in only two dimensions are trained. Two DMPs with the same canonical system are trained as foot trajectory generators in the x and z directions. The gait of the humanoid robot can be easily adjusted by modulating the DMP parameters. Fig. 4 shows the results of the trained foot trajectory generator and presents the foot trajectories for four steps of the robot. The blue solid lines represent the demonstration trajectory, and the red dashed lines represent the output of the foot trajectory generator.

Fig. 5 shows foot trajectories with different step lengths, lift heights and slopes: the blue solid lines represent the demonstration trajectory, and the dashed lines show different outputs of the foot trajectory generator.

B. ADAPTABILITY IMPROVEMENTS

A person walking on a slope adjusts his or her gait pattern and body attitude to maintain balance during walking. In this study, the vestibulospinal reflex mechanism is imitated and incorporated into the proposed method to generate adaptive

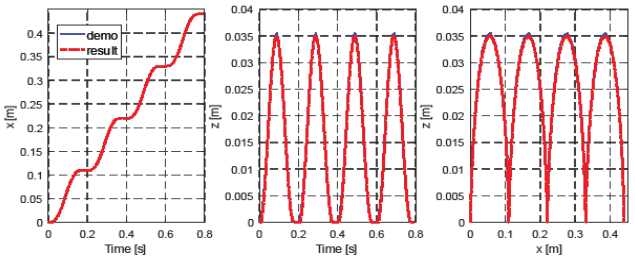


FIGURE 4. Foot trajectories for four steps.

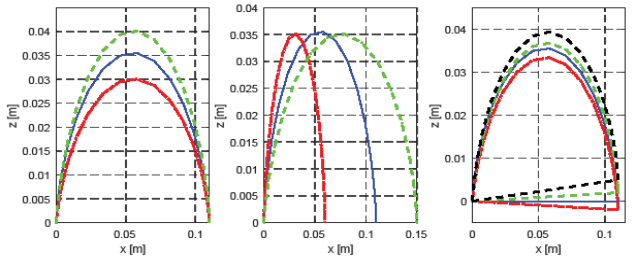


FIGURE 5. Foot trajectories after modulation of the parameters.

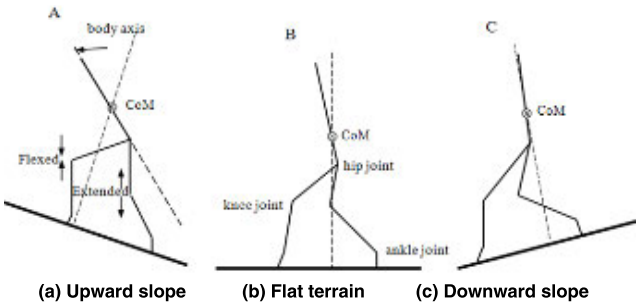


FIGURE 6. Body attitude adjustment during walking on different types of terrain.

walking motions for sloped terrain. As shown in Fig. 6, during walking up a slope, the feedback loop is designed to drive the CoM to move ahead along the slope and to reduce the gait length to maintain dynamic stability. Conversely, during walking down a slope, the CoM will be moved backward along the slope, and the gait length will be increased to prevent loss of balance.

Hence, it is necessary to collect sensory information corresponding to the surface conditions. It was experimentally verified by that body attitude sensing is effective for detecting the ground environment during walking. Hence, as Fig. 7 shows, body attitude information is used in this work as feedback to adjust the generated foot and CoM trajectories in an online manner to improve the walking quality. The body attitude angle θ_{pitch} can be determined using the robot's acceleration sensor.

C. INVERSE KINEMATICS

The motion engine is designed as shown in the dotted line in the Fig.8, where T^{ref} and T^{real} represent the reference trajectory and the actual trajectory, and J represents the Jacobian matrix, θ^{real} is the actual joint angle.

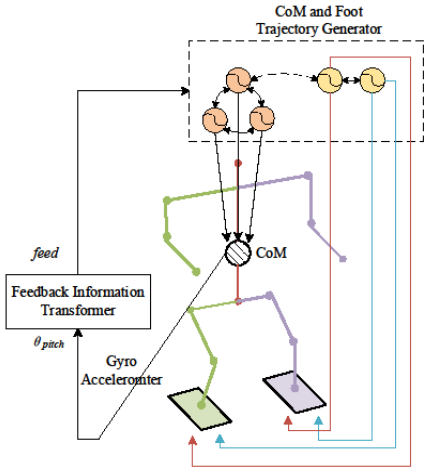


FIGURE 7. Structure of the feedback loop.

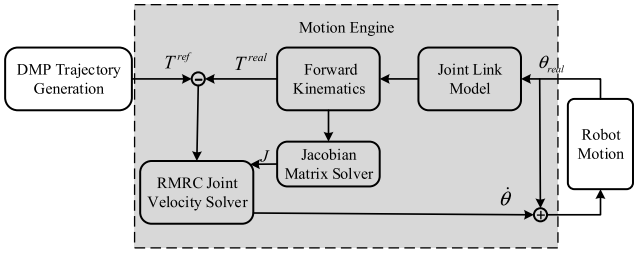


FIGURE 8. Motion engine for the biped walking.

The relationship between the workspace and the joint space can be described as follows:

$$L = f(\alpha) \tag{6}$$

where $L = [L_1, \cdots, L_M]^T$ represents the position and pose at the end of a movement and each α_i in $\alpha = [\alpha_1, \cdots, \alpha_N]^T$ represents the displacement or angle of link i with respect to link $i-1$.

The RMRC (Resolved motion rate control) method [39] is used to control the angular velocities of the joints. The differential kinematics and inverse differential kinematics of the robot can be described as follows:

$$\dot{L} = Q(\alpha) \cdot \dot{\alpha} \tag{7}$$

$$\dot{\alpha} = Q^{-1}(\alpha) \cdot \dot{L} \tag{8}$$

where $Q(\alpha)$ is the Jacobian matrix and $Q^{-1}(\alpha)$ is its inverse matrix.

When N > M, the robotic system is redundant. The $Q^{-1}(\alpha)$ does not exist when $Q(\alpha)$ is not full; instead, it is replaced by the pseudo-inverse matrix [40]:

$$Q^{\#} = Q^T (QQ^T)^{-1} \tag{9}$$

To avoid singularity problems, the pseudo-inverse matrix can be represented by means of the least squares method [41] as follows:

$$Q^{\#} = \left(Q^T Q + l^2 I\right)^{-1} Q^T \tag{10}$$

TABLE 1. Parameter values of the DMP models.

Parameter	Value
α_v	25
β_v	6.25
α_u	8.25
N	20

where l is a damping coefficient introduced to ensure that the Jacobian matrix is of full rank.

III. SIMULATIONS AND REAL EXPERIMENTS

A. EXPERIMENTAL SETUP

In this section we report the set of experiments using NAO robot under different conditions in order to verify the performance of the proposed control scheme. This work considers only the 10 DoFs of the robot's two legs. In this work, the adaptive characteristics of the trajectories were mapped by a motion engine between the workspace and the joint space. The required joint control signals for the proportional derivative servo actuators in the joints were calculated via inverse kinematics. The parameter evolution process was first conducted in the Webots and then transferred to real NAO robot. Thus, the optimal parameters were obtained on the simulated NAO robot.

B. WALKING PATTERN EVOLUTION

1) PARAMETERS OF TRAJECTORY LEARNING

Before DMPs can be used to implement learning by demonstration and to generate new trajectories, certain parameters must be determined. When $\beta_v = \alpha_v/4$ is satisfied, the system is globally stable with an attractor point g. The number of exponential basis functions, N, affects the trajectory learning process. A larger N corresponds to a DMP with a better output trajectory but a longer learning time; thus, a compromise between these two factors is required. Once the learning parameters have been determined, the CoM trajectory generator and the foot trajectory generator share these parameters. Table I shows the parameters determined based on simulations.

2) FEEDBACK PARAMETERS

NSGA-II [42] was adopted to optimize the feedback parameters for the proposed system to realize walking on flat terrain. This simulation was conducted using the Webots simulator. Binary tournament selection, intermediate crossover and Gaussian mutation methods were applied in this work. Two objective functions, $fitness_{atti}$ and $fitness_{dis}$, were designated to evolve the CoM and foot trajectories:

$$fitness_{atti} = \sum \left(|\theta_{pitch}| + |\theta_{roll}|\right) \tag{11}$$

$$fitness_{dis} = -dis \tan ce_x(robot) \tag{12}$$

where $fitness_{atti}$ reflects the stability of the robot throughout the entire walking process and $fitness_{dis}$ represents the distance the robot moves. In the simulation experiment, the population size and the maximum number of generations were set to 50 and 100, respectively.

C. INCLINED TERRAIN ADAPTIVE WALKING

1) REGULAR SLOPE TERRAIN ADAPTIVE WALKING

The walking terrain include three stages: flat surface, uphill and downhill segments, the slope terrain featuring variable inclination angles with unknown and arbitrary slopes of up to 12°. To realize adaptive walking, the body-attitude-based reflex method was used in this work. The body attitude θ can be estimated by applying the gyroscope and accelerometer.

While walking uphill, the robot should lean forward to avoid slippage, whereas while walking downhill, the robot should lean slightly back to maintain its balance. Thus, the feedback coupled to the DMP corresponding to the CoMx trajectory was designed as follows:

$$feed_{CoM_x} = K_{CoM_x} \times \theta \tag{13}$$

where K_{CoM_x} is the feedback gain that is used to modulate the CoM$_x$ trajectory along the sloped terrain to prevent overturning and slippage.

The feedback formulas coupled to the DMPs corresponding to the x-direction and z-direction foot trajectories, representing the step length and height, respectively, are expressed as follows:

$$feed_{footx} = -K_{footx} \times \theta \tag{14}$$

$$feed_{footz} = -K_{footz} \times \theta \tag{15}$$

where K_{footx} and K_{footz} are the feedback gains in the two directions.

The foot trajectory does not include information concerning the rotation angle of the robot's foot. To avoid friction between the foot and the ground during movement and produce a human-like gait, the rotation angle of the robot foot was designed as follows:

$$\omega_{\sup Foot} = \left[0,\ K_{\sup},\ 0\right]^T \tag{16}$$

$$\omega_{s\omega Foot} = \left[\omega_x \cdot l(t),\ \omega_y \cdot l(t) + K_{s\omega} \cdot \theta,\ 0\right]^T \tag{17}$$

where $\omega_{\sup Foot}$ and $\omega_{s\omega Foot}$ are the rotation angles of the robot's supporting and swing feet, respectively; ω_x and ω_y are constants that denote the maximum values around the x and y directions, respectively; $l(t)$ is a time function ranging from 0 to 1; and $K_{\sup}$ and $K_{s\omega}$ are the feedback gains for the supporting and swing feet, respectively.

As described above, the feedback information for the CoM generator, $feed_{CoM_x}$, is intended to adjust CoM$_x$ on inclined terrain to prevent slippage and overturning during walking. The feedback information for the foot trajectory generator, $feed_{\{footx,footz\}}$, is used to change the length and height of the step and to imitate a human's ability to adapt to changes in ground slope. The feedback information for the foot rotation angle, $[\omega_{\sup Foot}, \omega_{s\omega Foot}]$, is used to make the robot's gait more human-like and to adapt to changes in ground slope.

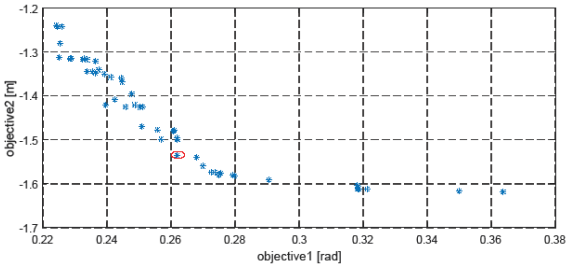


FIGURE 9. Pareto front in generation 150.

TABLE 2. Feedback parameter values.

Parameter	Value
K_{CoM_x}	0.9795
K_{footx}	0.7793
K_{footz}	1.4321
$K_{\sup}$	0.5681
$K_{s\omega}$	0.5213

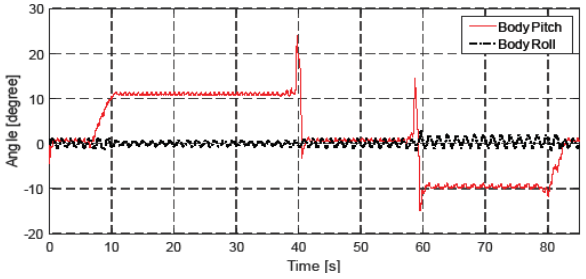
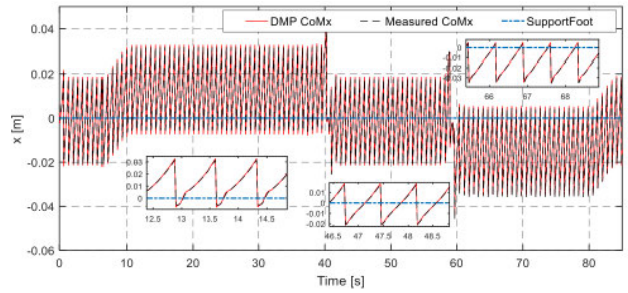


FIGURE 10. θ_{pitch} and θ_{roll} planes during slope terrain walking.

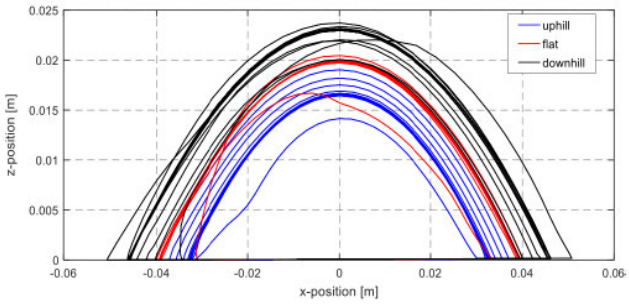
At the beginning stage of the evolution process, the NAO robot fell when it started uphill. After approximately 40 generations, NAO could achieve stable walking gait. Fig. 9 shows the evolution results of the 150th generation. Table 2 shows the parameters for one solution selected from among the best solutions on the Pareto front. These parameters for the neural oscillators remained the same in the subsequent experiments, independent of the terrain.

The experiment lasted for 86 s. Fig. 10 shows the changes in the body pitch and roll angles during the experiment. The red solid line represents the body pitch, and the black dotted line represents the body roll.

Fig. 11 shows the results of introducing the body posture as feedback; the CoM and foot trajectories can be adjusted online in real time. During uphill walking, the CoM moves ahead along the slope in the x direction, whereas it moves backward during downhill walking. Simultaneously, the foot trajectory is generated online, and the heights and lengths of the steps are adjusted in accordance with the slope. During uphill walking, the step height and length decrease in



(a) CoM trajectory



(b) Foot trajectory in the *x*-*z* plane

FIGURE 11. CoM and foot trajectories generated online.

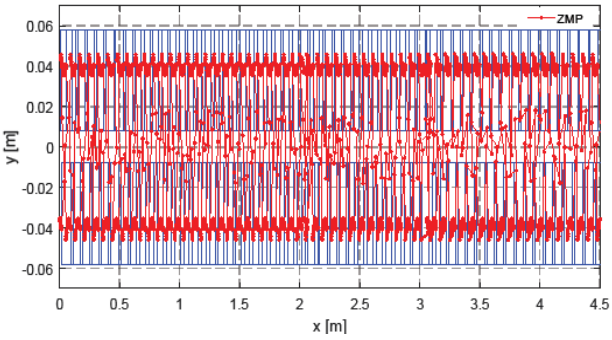


FIGURE 12. ZMP distribution during walking on sloped terrain.

accordance with the feedback information to prevent sliding, and during downhill walking, they increase to prevent overturning.

The ZMP distribution during the sloped terrain walking as shown in Fig. 12. The blue line shows the robot's foot support area. During the walking process, the ZMP remains within the robot's foot support area. Fig. 13 shows the joint control signals of the right leg, and the snapshots of a successful walking pattern on sloped terrain as shown in Fig. 14. The experimental simulation results indicate that the control system is effective and achieves stable walking during the transition from a level plane onto a sloped plane with a grade of up to 27.8%.

2) ADAPTIVE WALKING ON TERRAIN WITH A CHANGING SLOPE

In the second experiment, NAO robot was directed to walk across a inclined terrain with varying slope angles of 4°, 6°, 8°, 10° and 12°. The design of the feedback loop was the

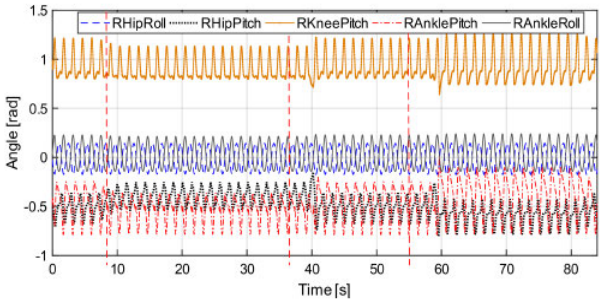


FIGURE 13. Joint control signals for the right leg.

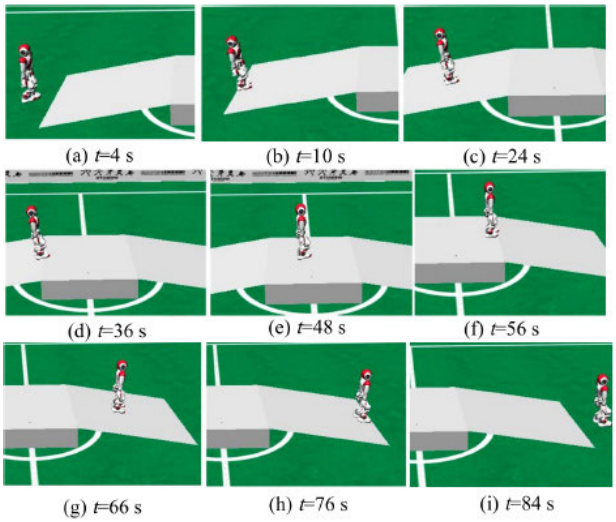


FIGURE 14. Snapshots of the slope terrain walking.

same as that used in the previous experiment. The CoM and foot generators were modulated online to generate adaptive trajectories based on the slope changes. Fig. 15 shows screenshots of the entire experiment. Fig. 16 shows the changes in the body attitude angles on the slope.

3) COMPARISON WITH THE JOINT-SPACE DMP METHOD

In current joint-space DMP control methods, DMP models are generally used to learn and generate angular control signals that serve as references for joint servo motors. The whole control architecture is set as shown in Fig. 17; the main central processing unit module, i.e., the DMP network, is used for the online generation of joint control signals, and the components indicated by dashed lines form the feedback loop that is used to modulate the DMP outputs to dynamically achieve adaptive walking on irregular ground. While the robot is walking, the control of each degree of freedom requires one DMP unit, and there is a total of ten joints related to the locomotion of both legs. Thus, the DMP network consists of ten DMP units, as shown in Fig. 18. This joint-space DMP method can be used to successfully achieve bipedal walking on flat ground. One set of the joint control signals for the left leg is shown in Fig. 19.

For a comparative adaptive walking experiment, consistent settings were adopted for the experimental environment, and

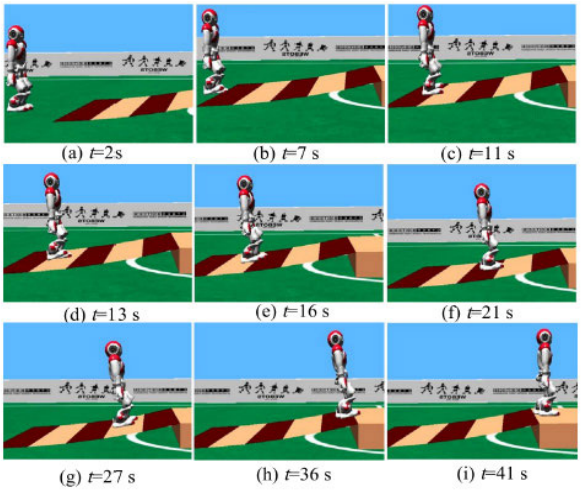


FIGURE 15. Snapshots of the walking process on the changing slope.

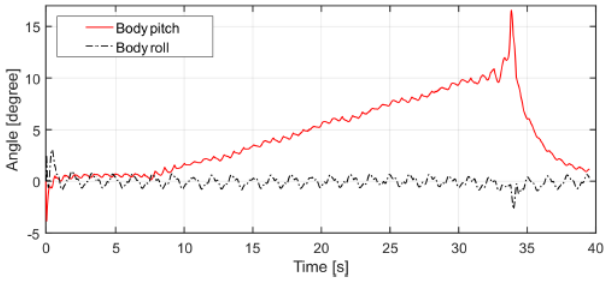


FIGURE 16. Body attitude in the pitch and roll planes.

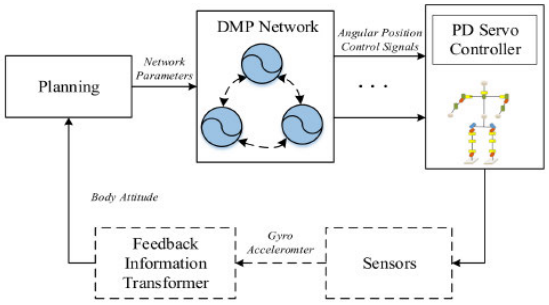


FIGURE 17. Joint-space DMP control architecture for bipedal walking.

the feedback loop was designed in a manner similar to that for the workspace DMP method:

$$feed_{DoFi} = K_{DoFi} \times \theta, \quad i = 1, 2, \cdots, 10 \tag{18}$$

where $feed_{DoFi}$ and K_{DoFi} represent the feedback term and feedback gain coefficients, respectively, for the i-th degree of freedom in the DMP network.

However, with the joint-space DMP method, the robot failed to complete the slope-adaptive experiment. After the NSGA-II evolution was complete, the robot still could not achieve stable adaptive walking on the slope. In the joint-space method, the CoM of the robot changes unpredictably, which may lead to abnormal situations. In addition, the robot

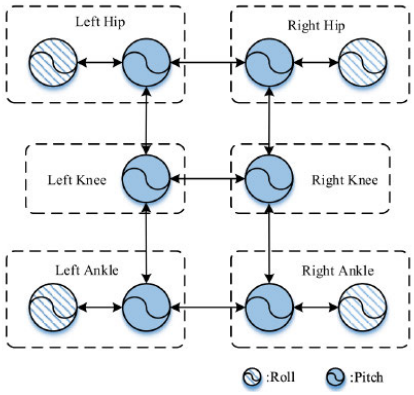


FIGURE 18. Structure of the DMP network.

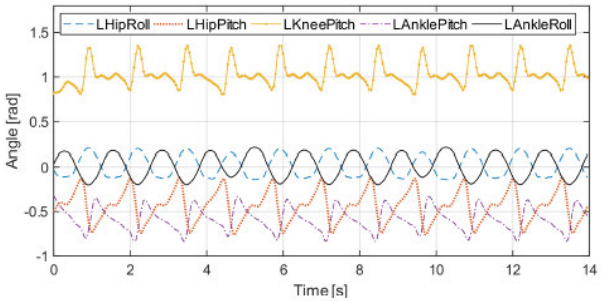


FIGURE 19. Control signals of the left leg.

will shake more when walking. Thus, compared with the joint-space DMP method, the proposed workspace DMP method is more effective. Fewer DMP units and parameters are required, meaning that less time is needed for evolution, and the CoM trajectory of the robot is intuitively controlled to prevent the robot from sliding and overturning. Thus, adaptive walking is successfully realized.

4) REAL ADAPTIVE WALKING EXPERIMENT

One experiment was designed to test the robot's performance with no prior knowledge of the terrain. In this experiment, the robot was made to walk on an elastically deformable platform consisting of three stages: two stages of inclined terrain (inclined surfaces with slopes of approximately 7°) with a flat connecting stage. Since the platform was elastically deformable, with each step of the robot on the platform, the platform would deform, causing large disturbances to the robot. With the presented method, as the robot walked across the inclined terrain, body-attitude-based feedback was used to automatically adjust the generated CoM and foot trajectories. During uphill walking, the CoM moved ahead along the slope in the x direction, whereas it moved backward during downhill walking. Simultaneously, the height and length of the foot trajectory were adjusted in accordance with the slope. Side-view snapshots of the experiment are shown in Fig. 20, illustrating that the robot could adjust its walking steps to reduce the influence of disturbances and successfully walk up and down the platform.

By incorporating feedback information, the basic walking pattern evolved via the EA was automatically adjusted;

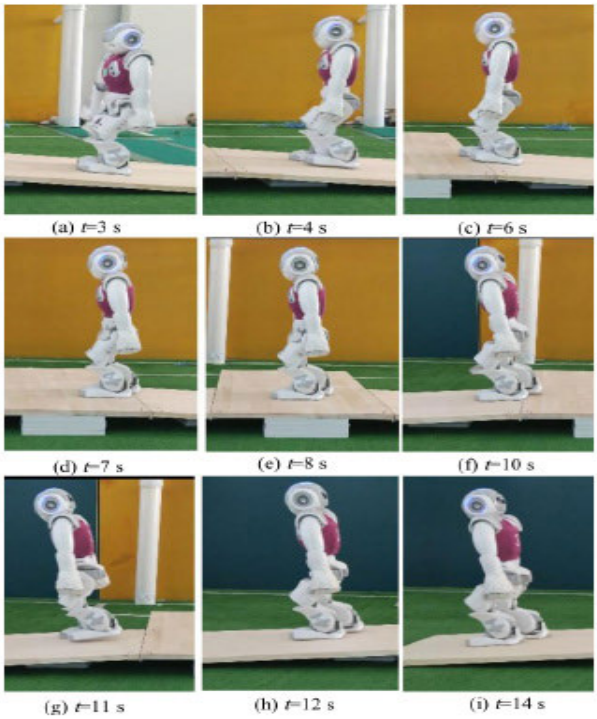
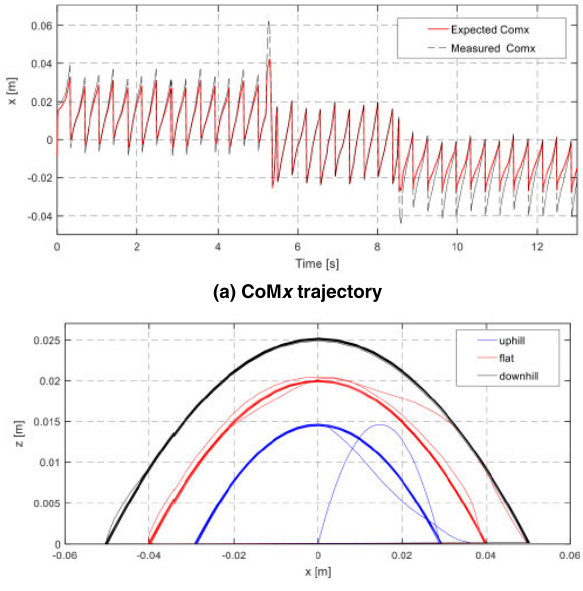


FIGURE 20. Snapshots of an adaptive walking experiment on sloped terrain.



(a) CoM*x* trajectory

(b) Foot trajectories in the *x*-*z* plane

FIGURE 21. CoM and foot trajectories generated online during the experiment.

consequently, the robot could autonomously adapt to the terrain conditions. Fig. 21 shows the CoM and foot trajectories generated online in the real experiment.

Fig. 22 shows the ZMP distribution during walking on sloped terrain. As observed, the ZMP was distributed within the supporting polygon of the robot's foot except during the phase of switching the supporting and swing legs. The differences between the Webots simulations and the experiments were caused by measurement noise, offsets and uncertainties

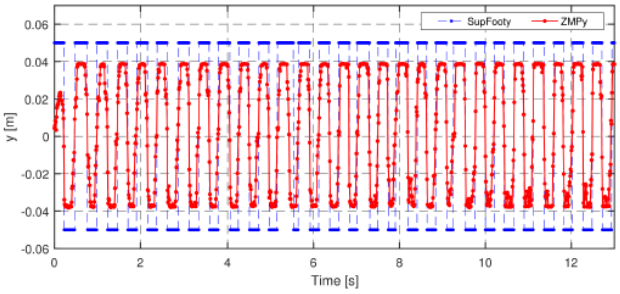


FIGURE 22. ZMP distribution.

in the robot model, including joint motor wear and slippage between the feet of the NAO robot and the walking surface. The NAO robot staggered only slightly during the terrain transitions. The experimental results show that the generated trajectories were viable and applicable.

IV. CONCLUSION AND FUTURE WORK

To enable a full-body humanoid robot to realize adaptive walking, a DMP-based workspace trajectory generation method is proposed. In the proposed locomotion control method, two online trajectory generators based on DMPs are used that allow bipedal robots to overcome the limitations of traditional bipedal locomotion control methods. The use of DMPs for planning workspace trajectories simplifies the complexity of trajectory planning and enables learning directly from a human walking gait. The proposed control strategy mimics the vestibular reflexes of human beings and includes multiple feedback loops to prevent robots from overturning and slipping. The proposed control system's performance was verified through both simulations and practical experiments with the NAO robot.

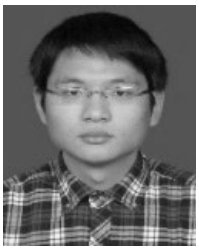
Future tasks will involve extending the feedback controller design to consider unknown external disturbances and more varied types of terrain. Moreover, research on the adaptation of the parameters of the walking controller will be investigated with the goal of developing mathematical models that will allow the control system to automatically compute the optimal parameters in any direction and at any prescribed velocity without requiring the EA-based optimization procedure to be executed. It will be important to develop practical strategies for automatically determining the feedback gains based on environmental sensory information to address increasingly challenging walking environments by focusing on adapting the robot's gait pattern to irregular ground, compensating for external perturbations and stepping over obstacles.

REFERENCES

[1] C. Liu, T. Zhang, C. Zhang, M. Liu, and Q. Chen, "Foot placement compensator design for humanoid walking based on discrete control Lyapunov function," *IEEE Trans. Syst., Man, Cybern. Syst.*, pp. 1–10, 2019.
[2] L. Yang, Z. Liu, and Y. Chen, "Energy efficient walking control for biped robots using interval type-2 fuzzy logic systems and optimized iteration algorithm," *ISA Trans.*, vol. 87, pp. 143–153, Apr. 2019.
[3] V. De-León-Gómez, Q. Luo, A. Kalouguine, J. A. Pámanes, Y. Aoustin, and C. Chevallereau, "An essential model for generating walking motions for humanoid robots," *Robot. Auto. Syst.*, vol. 112, pp. 229–243, Feb. 2019.

[4] S. Ding, W.-H. Chen, K. Mei, and D. J. Murray-Smith, "Disturbance observer design for nonlinear systems represented by input–output models," *IEEE Trans. Ind. Electron.*, vol. 67, no. 2, pp. 1222–1232, Feb. 2020.
[5] H.-K. Shin and B. K. Kim, "Energy-efficient gait planning and control for biped robots utilizing the allowable ZMP region," *IEEE Trans. Robot.*, vol. 30, no. 4, pp. 986–993, Aug. 2014.
[6] K. Hu, C. Ott, and D. Lee, "Learning and generalization of compensative zero-moment point trajectory for biped walking," *IEEE Trans. Robot.*, vol. 32, no. 3, pp. 717–725, Jun. 2016.
[7] H.-K. Shin and B. K. Kim, "Energy-efficient gait planning and control for biped robots utilizing vertical body motion and allowable ZMP region," *IEEE Trans. Ind. Electron.*, vol. 62, no. 4, pp. 2277–2286, Apr. 2015.
[8] Q. Huang, K. Yokoi, S. Kajita, K. Kaneko, H. Arai, N. Koyachi, and K. Tanie, "Planning walking patterns for a biped robot," *IEEE Trans. Robot. Autom.*, vol. 17, no. 3, pp. 280–289, Jun. 2001.
[9] M. Vukobratovic, A. A. Frank, and D. Juricic, "On the stability of biped locomotion," *IEEE Trans. Biomed. Eng.*, vol. BME-17, no. 1, pp. 25–36, Jan. 1970.
[10] M. Vukobratovic and B. Borovac, "Zero-moment point—Thirty five years of its life," *Int. J. Hum. Robot.*, vol. 1, no. 1, pp. 157–173 2004.
[11] I.-W. Park, J.-Y. Kim, J. Lee, and J.-H. Oh, "Online free walking trajectory generation for biped humanoid robot KHR-3(HUBO)," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2006, pp. 1231–1236.
[12] U. Seven, T. Akbas, K. C. Fidan, and K. Erbatur, "Bipedal robot walking control on inclined planes by fuzzy reference trajectory modification," *Soft Comput.*, vol. 16, no. 11, pp. 1959–1976, Jun. 2012.
[13] Z. Yu, X. Chen, Q. Huang, W. Zhang, L. Meng, W. Zhang, and J. Gao, "Gait planning of omnidirectional walk on inclined ground for biped robots," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 46, no. 7, pp. 888–897, Jul. 2016.
[14] Y. Wu, B. Jiang, and N. Lu, "A descriptor system approach for estimation of incipient faults with application to high-speed railway traction devices," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 49, no. 10, pp. 2108–2118, Oct. 2019.
[15] Y. Wu, B. Jiang, and Y. Wang, "Incipient winding fault detection and diagnosis for squirrel-cage induction motors equipped on CRHtrains," *ISA Trans.*, 2020, doi: 10.1016/j.isatra.2019.09.020.
[16] L. Fang, L. Ma, S. Ding, and D. Zhao, "Finite-time stabilization for a class of high-order stochastic nonlinear systems with an output constraint," *Appl. Math. Comput.*, vol. 358, pp. 63–79, Oct. 2019.
[17] Y. Yuan, Z. Li, T. Zhao, and D. Gan, "DMP-based motion generation for a walking exoskeleton robot using reinforcement learning," *IEEE Trans. Ind. Electron.*, vol. 67, no. 5, pp. 3830–3839, May 2020.
[18] R. S. Sharma, S. Shukla, H. Karki, A. Shukla, L. Behera, and K. S. Venkatesh, "DMP based trajectory tracking for a nonholonomic mobile robot with automatic goal adaptation and obstacle avoidance," in *Proc. Int. Conf. Robot. Autom. (ICRA)*, May 2019, pp. 8613–8619.
[19] S. Qiu, W. Guo, D. Caldwell, and F. Chen, "Exoskeleton online learning and estimation of human walking intention based on dynamical movement primitives," *IEEE Trans. Cognit. Develop. Syst.*, to be published.
[20] A. Jan Ijspeort, J. Nakanishi, and S. Schaal, "Learning rhythmic movements by demonstration using nonlinear oscillators," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, vol. 1, Oct. 2002, pp. 958–963.
[21] A. J. Ijspeert, J. Nakanishi, H. Hoffmann, P. Pastor, and S. Schaal, "Dynamical movement primitives: Learning attractor models for motor behaviors," *Neural Comput.*, vol. 25, no. 2, pp. 328–373, Feb. 2013.
[22] S. Schaal, "Dynamic movement primitives-A framework for motor control in humans and humanoid robotics," in *Adaptive Motion of Animals and Machines*, H. Kimura, K. Tsuchiya, A. Ishiguro, and H. Witte, Eds. Tokyo, Japan: Springer, 2006.
[23] A. Ude, A. Gams, T. Asfour, and J. Morimoto, "Task-specific generalization of discrete and periodic dynamic movement primitives," *IEEE Trans. Robot.*, vol. 26, no. 5, pp. 800–815, Oct. 2010.
[24] K. Mülling, J. Kober, O. Kroemer, and J. Peters, "Learning to select and generalize striking movements in robot table tennis," *Int. J. Robot. Res.*, vol. 32, no. 3, pp. 263–279, Jan. 2013.
[25] J. Umlauft, D. Sieber, and S. Hirche, "Dynamic movement primitives for cooperative manipulation and synchronized motions," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 766–771.
[26] D.-H. Park, H. Hoffmann, P. Pastor, and S. Schaal, "Movement reproduction and obstacle avoidance with dynamic movement primitives and potential fields," in *Proc. Hum. 8th IEEE-RAS Int. Conf. Hum. Robots*, Dec. 2008, pp. 91–98.
[27] H. Hoffmann, P. Pastor, D.-H. Park, and S. Schaal, "Biologically-inspired dynamical systems for movement generation: Automatic real-time goal adaptation and obstacle avoidance," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2009, pp. 2587–2592.
[28] A. Gams, A. J. Ijspeert, S. Schaal, and J. Lenarčič, "On-line learning and modulation of periodic movements with nonlinear dynamical systems," *Auto. Robots*, vol. 27, no. 1, pp. 3–23, May 2009.
[29] J. Kober and J. Peters, "Learning motor primitives for robotics," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2009, pp. 2112–2118.
[30] J. Peters and S. Schaal, "Reinforcement learning of motor skills with policy gradients," *Neural Netw.*, vol. 21, no. 4, pp. 682–697, May 2008.
[31] Z. Li, T. Zhao, F. Chen, Y. Hu, C.-Y. Su, and T. Fukuda, "Reinforcement learning of manipulation and grasping using dynamical movement primitives for a humanoidlike mobile manipulator," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 1, pp. 121–131, Feb. 2018.
[32] A. Ude, B. Nemec, T. Petric, and J. Morimoto, "Orientation in Cartesian space dynamic movement primitives," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 2997–3004.
[33] J. Nakanishi, J. Morimoto, G. Endo, G. Cheng, S. Schaal, and M. Kawato, "A framework for learning biped locomotion with dynamical movement primitives," in *Proc. 4th IEEE/RAS Int. Conf. Hum. Robots*, vol. 2, Nov. 2004, pp. 925–940.
[34] J. Morimoto, G. Endo, J. Nakanishi, and G. Cheng, "A biologically inspired biped locomotion strategy for humanoid robots: Modulation of sinusoidal patterns by a coupled oscillator model," *IEEE Trans. Robot.*, vol. 24, no. 1, pp. 185–191, Feb. 2008.
[35] P. Pastor, H. Hoffmann, T. Asfour, and S. Schaal, "Learning and generalization of motor skills by learning from demonstration," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2009, pp. 763–768.
[36] J. Rosado, F. Silva, V. Santos, and A. Amaro, "Adaptive robot biped locomotion with dynamic motion primitives and coupled phase oscillators," *J. Intell. Robot. Syst.*, vol. 83, nos. 3–4, pp. 375–391, Jan. 2016.
[37] D. Luo, X. Han, Y. Ding, Y. Ma, Z. Liu, and X. Wu, "Learning push recovery for a bipedal humanoid robot with dynamical movement primitives," in *Proc. IEEE-RAS 15th Int. Conf. Hum. Robots (Humanoids)*, Nov. 2015, pp. 1013–1019.
[38] A. Böckmann and T. Laue, "Kick motions for the NAO robot using dynamic movement primitives," 2016, *arXiv:1606.00600*. [Online]. Available: http://arxiv.org/abs/1606.00600
[39] Y. Umetani and K. Yoshida, "Resolved motion rate control of space manipulators with generalized jacobian matrix," *IEEE Trans. Robot. Autom.*, vol. 5, no. 3, pp. 303–314, Jun. 1989.
[40] S. R. Buss, "Introduction to inverse kinematics with jacobian transpose, pseudoinverse and damped least squares methods," *IEEE J. Robot. Automat.*, vol. 17, nos. 1–19, p. 16, Apr. 2004.
[41] G. Marani, J. Kim, J. Yuh, and W. Kyun Chung, "Algorithmic singularities avoidance in task-priority based controller for redundant manipulators," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, vol. 4, Oct. 2003, pp. 3570–3574.
[42] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Trans. Evol. Comput.*, vol. 6, no. 2, pp. 182–197, Apr. 2002.

CHENGJU LIU received the Ph.D. degree in control theory and control engineering from Tongji University, Shanghai, China, in 2011. From October 2011 to July 2012, she was a Research Associate with the BEACON Center, Michigan State University, East Lansing, USA. From March 2011 to June 2013, she was a Postdoctoral Researcher with Tongji University where she is currently a Professor with the College of Electrical and Information Engineering. She is also the Team Leader of the TJArk Robot Team, Tongji University. Her research interests include intelligent control, motion control of legged robots, and evolutionary computation.



WANDONG GENG graduated in 2018. He is currently pursuing the master's degree in control theory and control engineering with Tongji University. His field of study is the locomotion of biped robot.

MING LIU (Senior Member, IEEE) received the B.A. degree in automation from Tongji University, Shanghai, China, in 2005, and the Ph.D. degree from the Department of Mechanical Engineering and Process Engineering, ETH Zürich, Zürich, Switzerland, in 2013. He was a Visiting Scholar with the University of Erlangen-Nuremberg, Erlangen, Germany, and the Fraunhofer Institute IISB, Erlangen. He is currently an Assistant Professor with the Department of Electronic and Computer Engineering, The Hong Kong University of Science and Technology, Hong Kong. His current research interests include autonomous mapping, visual navigation, topological mapping, and environment modeling.

QIJUN CHEN (Senior Member, IEEE) received the B.S. degree in automatic control from the Huazhong University of Science and Technology, Wuhan, China, in 1987, the M.S. degree in control engineering from Xi'an Jiaotong University, Xi'an, China, in 1990, and the Ph.D. degree in control theory and control engineering from Tongji University, Shanghai, China, in 1999. He was a Visiting Professor with the University of California at Berkeley, Berkeley, CA, USA, in 2008. He is currently a Professor with the College of Electronic and Information Engineering, Tongji University. His current research interests include network-based control systems and robotics.