




RESEARCH ARTICLE

# A heuristic gait template planning and dynamic motion control for biped robots

Lianqiang Han<sup>1</sup> , Xuechao Chen<sup>1,\*</sup> , Zhangguo Yu<sup>1</sup>, Zhifa Gao<sup>1</sup> , Gao Huang<sup>2</sup>, Jintao Zhang<sup>1</sup>, Kenji Hashimoto<sup>3</sup> and Qiang Huang<sup>1</sup>

<sup>1</sup>School of Mechatronic Engineering, Beijing Institute of Technology, Beijing, China, <sup>2</sup>Faculty of Information Technology, Beijing University of Technology, Beijing, China, and <sup>3</sup>Department of Mechanical Engineering Informatics, Meiji University/Humanoid Robotics Institute (HRI), Waseda University, Kanagawa, Japan

\*Corresponding author. E-mail: [chenxuechao@bit.edu.cn](mailto:chenxuechao@bit.edu.cn)

Received: 27 June 2022; Revised: 9 October 2022; Accepted: 17 October 2022; First published online: 15 November 2022

Keywords: gait template, model-free, biped robot, dynamic motion control, joint torque control

## Abstract

Biped robots with dynamic motion control have shown strong robustness in complex environments. However, many motion planning methods rely on models, which have difficulty dynamically modifying the walking cycle, height, and other gait parameters to cope with environmental changes. In this study, a heuristic model-free gait template planning method with dynamic motion control is proposed. The gait trajectory can be generated by inputting the desired speed, walking cycle, and support height without a model. Then, the stable walking of the biped robot can be realized by foothold adjustment and whole-body dynamics model control. The gait template can be changed in real time to achieve gait flexibility of the biped robot. Finally, the effectiveness of the method is verified by simulations and experiments of the biped robot BHR-B2. The research presented here helps improve the gait transition ability of biped robots in dynamic locomotion.

## 1. Introduction

Compared with wheeled robots, biped robots can better adapt to rugged environments, such as gullies and stairs. Robots will help people repeat boring tasks after increasing the ability of dexterous operation [1]. Therefore, people hope bipedal robots can enter working environments to help them complete dangerous tasks or explore isolated areas [2–5]. The Atlas robot from Boston dynamics has become a famous representative of the application of robotics in engineering [6]. However, public information on how its algorithm works is still lacking. The motion planning and control algorithm of biped robots has been widely discussed in recent years [7, 8]. Currently, very few biped robots can move dynamically and stably outdoors [9–11]. One of the critical influencing factors is that dynamic motion planning and trajectory correction significantly impact environmental adaptability. For instance, in a complex environment, it is sometimes necessary for the robot to adjust the support height to proceed, adjust the step frequency to maintain stability, or adjust the foot lifting height to avoid a collision. Therefore, when facing a complex dynamic environment, the robot's dynamic modified gait plays an important role in its balance and stability. In general, the existing algorithms, which can be divided into model-based and model-free algorithms, have some limitations in dynamically adjusting the above parameters.

The principle of the model-based method is to consider the dynamic characteristics of a model and let the biped robot simulate this model to generate a state sequence. The most famous model-based approach is the gait generator based on the linear inverted pendulum (LIP) [12, 13], which has a constant height of the center of mass (CoM). By simplifying the biped robot into an LIP and planning the zero moment point (ZMP) trajectory, the CoM trajectory is then obtained. Later, to control the unstable divergent state, the divergent components of motion method was proposed, and experiments were successfully carried

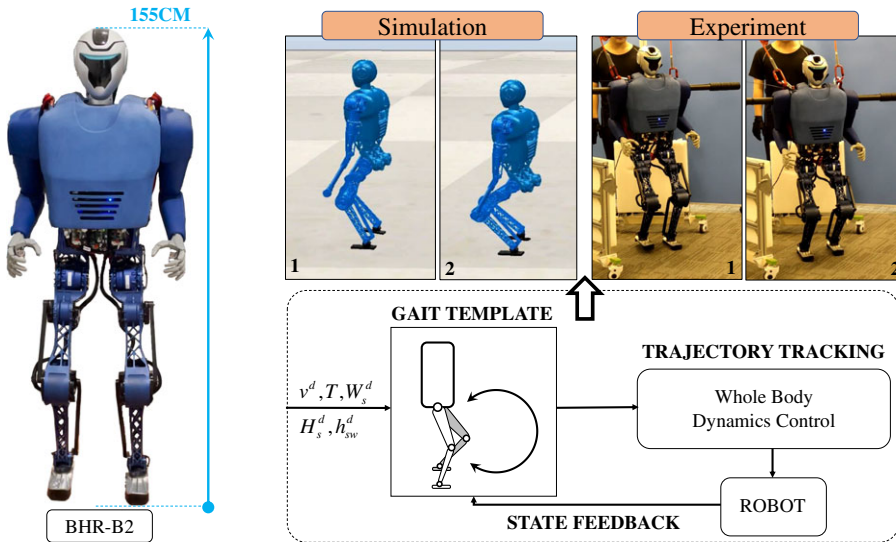
out on several robots [14]. However, when these methods resist disturbance, compliance controls need to be added to stabilize themselves [15]. Application scenarios are limited to flat indoor surfaces ground. Xiong and Ames proposed the hybrid-LIP model to introduce the two-foot support period [16] so that the underactuated biped robot (UBR) can achieve the stability even with significant disturbances. Similarly, the spring-loaded inverted pendulum model is also used [17]. However, using these simplified models results in the algorithm requiring the robot centroid to considerably upwards [18], which is challenging to satisfy this requirement for some biped robots due to their ability needs. With the development of nonlinear optimization in trajectory planning, trajectory optimization based on the whole-body model of a biped robot has been studied [19, 20]. In this method, a dynamic model of the biped robot's entire body is established, the motion library is generated by offline optimization, and the desired action is realized by real-time control. However, for a biped robot with more than 10 or 20 degrees of freedom (DoFs), optimization will consume a lot of time and is also sensitive to the initial state [21–23]. In addition, the offline gait library will have limitations for the environment that has not been established. More importantly, due to the property constraints of the model itself, changing parameters online such as walking cycle and height will cause model mismatch. Therefore, model-based trajectory planning has some limitations in real-time variable parameters.

The essence of the model-free method is to generate actions through heuristic strategies. The famous representative of this method is the set of three strategies used by Raibert to control hopping robots [24]. This method relies on the robot's swing and then cooperates with the upper body attitude control and foothold control to realize the movement of legged robots. This strategy is still used in the Atlas robot [25]. It also significantly impacts the dynamic motion planning and control of biped robots [26, 27]. Nonmodel-based trajectory planning can use fewer parameters to form rich gait motion. Yin et al. [28] proposed the SIMBICON method to realize the animation simulation of multiple actions of the robot by giving a particular joint state but did not consider the robot performance and dynamic model. Recently, artificial intelligence learning algorithm has made remarkable progress in the robotics field [29, 30]. This kind of method mainly involves setting heuristic rewards so that the robot can constantly make attempts and make mistakes to obtain the policy that generates the proper trajectory [31]. Relevant research focuses on Cassie and Digit robots with good performance and shows unexpected results of experiments in real environments [9, 32]. However, such research is still in the preliminary stage, and the high requirements for drivers and mechanical structure make it difficult to promote and popularize it. The biped robot completes the walking action by the heuristic method easily and quickly, but there is no unified framework for model-free motion planning including multiple gait parameters.

Therefore, to solve the problem that the gait parameters cannot be modified in real time to adapt to changing environments, we propose a heuristic model-free gait template for walking planning and control to establish a unified mathematical framework (as shown in Fig. 1). It consists of the following contributions.

- (1) The gait template does not depend on the model, which integrates gait parameters such as support height, walking cycle, foot lifting height, and walking speed to form a specific mathematical expression, which is convenient for biped robots to generate walking trajectories.
- (2) The gait template allows online trajectory generation and real-time correction and realizes the robot's motion following through whole-body dynamics control (WBC), which is conducive to the improvement of dynamic environment adaptability.
- (3) A large biped robot with integrated structure is developed, and the feasibility and effectiveness of the algorithm are verified on the robot.

The rest of this paper is arranged as follows: Section 2 introduces the design process of the gait template in detail. Section 3 shows the control method for realizing the gait trajectory. Section 4 presents the analysis of experiments and simulations, and Section 5 is a summary and discussion.



**Figure 1.** The BHR-B2 robot uses the trajectory generated by the gait template to move in the simulated environment and the experiment with dynamic motion control. On the left is the robot experimental platform. The total height of the robot is 155 cm.

## 2. Gait template

### 2.1. Walking trajectory design

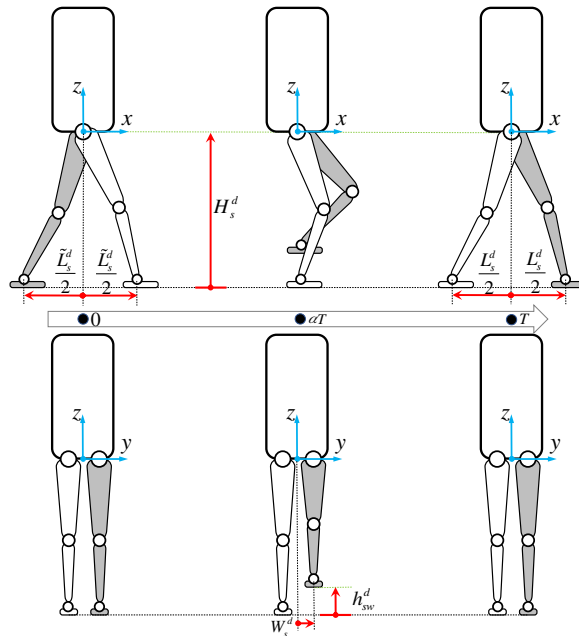
Inspired by natural human motion, we used a heuristic strategy to plan the movement of both legs. Because the period of time in which both legs are supporting weight during walking is small, that period of time is ignored by the gait template. We only consider the stage of alternating left and right supporting legs. We define one step as follows: the process of the leg from contact with the ground to swinging in the air and then contact with the environment. Figure 2 shows the movement of both legs during one step. Within each step, one leg must get to the ground, called the support leg (white color-filled) marked  $s$ , while the other leg completes the leg lifting and stepping down in the air to become a swing leg (gray color-filled) marked  $sw$ . Next, leg trajectory planning is described in detail.

As shown in Fig. 2, define the coordinate system  $xyz$  at the center of the connecting line between the two hip joints. The desired position of the support leg  $p_s^d = [x_s^d \ y_s^d \ z_s^d]^T$  and the desired position of the swing leg  $p_{sw}^d = [x_{sw}^d \ y_{sw}^d \ z_{sw}^d]^T$  are defined relative to this coordinate system. The superscript  $d$  indicates the desired value, and  $(\cdot)^T$  represents the transpose operator of the matrix. The walking cycle time is set to  $T$ , the desired height of the support leg is  $H_s^d$ , the desired width of the ankle relative to the  $y$  direction of the coordinate system is  $W_s^d$ , and the desired lifting height of the swinging leg is  $h_{sw}^d$ . The following plans are the three-dimensional coordinates of the ankle joint relative to the coordinate system. Then the angle of each joint can be obtained through the inverse kinematics of the leg. The ankle joint angle can also be obtained by adding the constraint that the footplate is always parallel to the horizontal plane. Next, we will explain in detail how the trajectory of the end of the leg is planned.

#### 2.1.1. Symbol definition

The trajectory of the rear of the leg is planned by cubic spline interpolation. To simplify the mathematical expression, we define some related variables here. We define the time vector for calculating position  $p(t)$ , velocity  $\dot{p}(t)$ , and acceleration  $\ddot{p}(t)$  in interpolation

$$\begin{aligned}
 p(t) &= [1 \quad t \quad t^2 \quad t^3 \quad t^4] \\
 \dot{p}(t) &= [0 \quad 1 \quad 2t \quad 3t^2 \quad 4t^3] \\
 \ddot{p}(t) &= [0 \quad 0 \quad 2 \quad 6t \quad 12t^2]
 \end{aligned}
 \tag{1}$$



**Figure 2.** Diagram of walking gait template. One-step action decomposition of the robot in the sagittal and coronal planes. The walking cycle is divided into two stages: raising the foot and stepping down.

Then, the interpolation coefficient calculation matrix function is defined as  $W$

$$W(t_0, t_1, t_2) = \begin{bmatrix} W_1(t_0, t_1) & \mathbf{0}^{4 \times 5} \\ \mathbf{0}^{4 \times 5} & W_1(t_2, t_1) \\ W_2(t_1) & -W_2(t_1) \end{bmatrix} \tag{2}$$

where  $t_0$  and  $t_2$  represent the initial time and the end time, respectively, while  $t_1$  takes a value between  $t_0$  and  $t_2$ .  $\mathbf{0}$  represents all zero matrices, and the superscript number represents the dimension. Matrix  $W_1$  is

$$W_1(a, b) = [p(a)^T \quad \dot{p}(a)^T \quad \ddot{p}(a)^T \quad p(b)^T]^T \tag{3}$$

where  $a$  and  $b$  are parameters. Matrix  $W_2$  is

$$W_2(t_1) = [\dot{p}(t_1)^T \quad \ddot{p}(t_1)^T]^T \tag{4}$$

when the following are given for the time interpolation: position  $\zeta_0$ , velocity  $v_0$ , and acceleration  $\psi_0$  corresponding to the initial time  $t_0$ ; the position  $\zeta_2$ , velocity  $v_2$ , and acceleration  $\psi_2$  corresponding to the end time  $t_2$ ; and the position  $\zeta_1$  corresponding to time  $t_1$ . With the initial parameter  $\chi_0 = [\zeta_0 \quad v_0 \quad \psi_0 \quad \zeta_1 \quad \zeta_2 \quad v_2 \quad \psi_2 \quad \zeta_1 \quad 0 \quad 0]^T$ , the cubic spline curve coefficient can be calculated

$$\xi = W(t_0, t_1, t_2)^{-1} * \chi_0 \tag{5}$$

where  $(\cdot)^{-1}$  is the square matrix inversion operator. Therefore, we use  $\xi(t_0, t_1, t_2, \chi_0) = [\xi_1 \quad \xi_2 \quad \dots \quad \xi_{10}]^T$  to represent the interpolation coefficients under all of the above initial conditions.

Thus, the position and speed reference trajectory of the leg in  $\kappa \in \{x, y, z\}$  direction can be expressed as

$$\begin{aligned} \kappa_s^d(t) &= \begin{cases} \xi(t_0, t_1, t_2, \chi_\kappa)^T * \begin{bmatrix} p(t)^T \\ \mathbf{0}^{5 \times 1} \end{bmatrix} & t_0 \leq t \leq t_1 \\ \xi(t_0, t_1, t_2, \chi_\kappa)^T * \begin{bmatrix} \mathbf{0}^{5 \times 1} \\ p(t)^T \end{bmatrix} & t_1 \leq t \leq t_2 \end{cases} \\ \dot{\kappa}_s^d(t) &= \begin{cases} \xi(t_0, t_1, t_2, \chi_\kappa)^T * \begin{bmatrix} \dot{p}(t)^T \\ \mathbf{0}^{5 \times 1} \end{bmatrix} & t_0 \leq t \leq t_1 \\ \xi(t_0, t_1, t_2, \chi_\kappa)^T * \begin{bmatrix} \mathbf{0}^{5 \times 1} \\ \dot{p}(t)^T \end{bmatrix} & t_1 \leq t \leq t_2 \end{cases} \end{aligned} \tag{6}$$

In the gait template, let  $\alpha$  represent the proportion of the leg lifting stage in the total walking cycle  $T$  in one step, and set  $t_0 = 0$ ,  $t_1 = \alpha T$ , and  $t_2 = T$ . When the initial condition  $\chi_\kappa$  is known, the position and velocity at the current time  $t$  can be obtained from Eq. (6). Therefore, we will only give initial parameter conditions in the following trajectory planning.

### 2.1.2. Support leg trajectory

The support leg will keep the torso of the robot moving horizontally and always keep the support height (height of the torso) unchanged. Ideally, the speed is zero from the static start to the final state. Therefore, the initial parameter of the support leg in the  $z$  direction is

$$\chi_z^s = [-H_s^d \ 0 \ 0 \ -H_s^d \ -H_s^d \ 0 \ 0 \ -H_s^d \ 0 \ 0]^T \tag{7}$$

The lateral motion ( $y$  direction) of the biped robot is mainly used for adjustment and stability in the whole walking process, so it remains unchanged in the gait template

$$\begin{aligned} \chi_y^s &= [a \ 0 \ 0 \ a \ a \ 0 \ 0 \ a \ 0 \ 0]^T \\ a &= (-1)^i W_s^d, i = \begin{cases} 0 & s \rightarrow \text{left} \\ 1 & s \rightarrow \text{right} \end{cases} \end{aligned} \tag{8}$$

In this formula, when the supporting leg is the left leg,  $i = 0$ , and when it is the right leg  $i = 1$ . For the motion in the  $x$  direction, we want the supporting leg to complete the swing motion from the current position to half of the desired step length

$$\chi_x^s = \left[ \frac{\tilde{L}_s^d}{2} \ 0 \ 0 \ 0 \ -\frac{L_s^d}{2} \ 0 \ 0 \ 0 \ 0 \ 0 \right]^T \tag{9}$$

where  $\tilde{L}_s^d$  represents the target step size of the previous step, further explaining the meaning of support leg  $x$ -direction position planning. The gait template set  $\varsigma_2 = -L_s^d/2$  moves the support leg in the opposite direction to push the upper body. In the gait template,  $\varsigma_1 = 0$ ; this way, after the swing leg is lifted to the desired height, the torso center of gravity passes through the support point. The acceleration of the trajectory can be obtained through the above interpolation process, but we do not show it here.

2.1.3. Swing leg trajectory

When the supporting leg moves, the swinging leg needs to lift and step down through the heuristic programming method. Similarly, in the y-direction, the swing leg remains constant

$$\chi_y^{sw} = [a \ 0 \ 0 \ a \ a \ 0 \ 0 \ a \ 0 \ 0]^T$$

$$a = (-1)^i W_s^d, i = \begin{cases} 0 & sw \rightarrow \text{left} \\ 1 & sw \rightarrow \text{right} \end{cases} \tag{10}$$

The movement of the swinging leg in the x-direction is opposite to that of the supporting leg. The goal is to find the next support point in the current movement trend. The interpolation initialization conditions are

$$\chi_x^{sw} = \left[ -\frac{\tilde{L}_s^d}{2} \ 0 \ 0 \ 0 \ \frac{L_s^d}{2} \ 0 \ 0 \ 0 \ 0 \ 0 \right]^T \tag{11}$$

When the torso center of gravity passes the support point, the gait template sets the lifting height to reach the desired value. Therefore, the initialization condition of interpolation in the z direction is

$$\chi_z^{sw} = [-H_s^d \ 0 \ 0 \ H \ -H_s^d \ 0 \ 0 \ H \ 0 \ 0]^T$$

$$H = -H_s^d + h_{sw}^d \tag{12}$$

The position and velocity of the swing leg can be obtained by Eq. (6).

2.2. Multistep switching strategy

The walking motion of a biped robot requires two legs to alternately support the torso. When the support leg (or swing leg) switches to the swing leg (or support leg) state, the initial conditions of the gait template will change. When the robot’s trajectory uses time-triggered switching, through the above description, the trajectory will switch smoothly because the end time of each step occurs in the desired state. However, biped robots use event triggering to switch leg states (such as changing immediately when the swinging leg touches the ground) for dynamic stability. This approach will lead to the touch-down time of biped robots being earlier than or later than the walking cycle *T*. Then, the initial condition of the next step will no longer be the expected state at the end of the previous step. Therefore, when the event is triggered, the gait template adopts the following switching rules:

- When the swing leg is going to be switched to the support leg, the instantaneous *desired state* at the switching time will become one of the initial conditions after it becomes the support leg.
- When the support leg is going to be switched to the swing leg, the instantaneous *actual state* at the switching time will become one of the initial conditions of the swing leg.

2.3. Dynamic adjustment method of the gait template

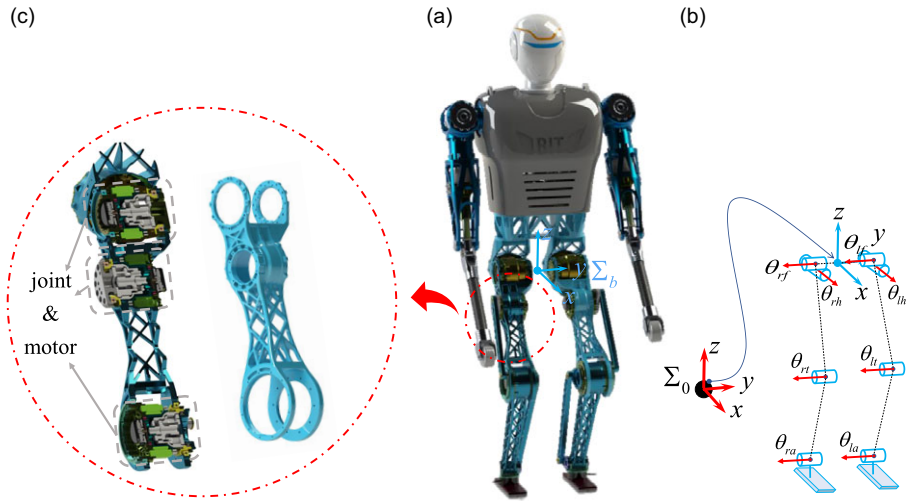
The gait template allows real-time dynamic adjustment of the desired speed and support height. When the desired speed  $v_x^d$  is changed, the reference step size will be updated simultaneously by the following formula:

$$L_s^d = T v_x^d \tag{13}$$

It is also possible to dynamically change the walking cycle time *T*. Then the interpolation time in the gait template will also change dynamically

$$t_0 = 0, t_1 = \alpha T, t_2 = T \tag{14}$$

In addition, the gait template allows the reference step width  $W_s^d$ , support height  $H_s^d$ , and foot lifting height  $h_s^d$  to change dynamically and then change the end time state in the interpolation initial conditions



**Figure 3.** Mechanical structure and degree of freedom configuration of the BHR-B2 robot. (a) 3D structure view of the robot. (b) The blue cylinder represents the driving joint of rotation, and the red arrow represents the positive direction of rotation of the joint with the right-handed system. (c) Integrated thigh structure and drive motor layout of the leg.

through Eqs. (7)–(12) to make a smooth transition to the desired state. These template parameters should be updated by the upper planner when switching the leg state after one step.

### 3. Trajectory tracking control

#### 3.1. Control walking speed through gait adjustment

The speed control is critical in the motion control of a heuristic dynamic biped robot, and the expected speed in the gait template can be obtained after the predicted trajectory is entirely executed according to Eq. (13). However, it is difficult for the robot to achieve this requirement with random environmental disturbances, so we also use the foothold adjustment method to gain further speed control. First, the adjusted increment is obtained according to the centroid velocity information

$$\Delta L = k_v (v_x - v_x^d) - k_a a_x \tag{15}$$

where  $k_v$  and  $k_a$  are gain coefficients. This increment will be compensated for gait adjustment at the desired step size  $L_s^d = L_s^d + \Delta L$ . The robot’s lateral (y direction) foothold compensation is carried out in the same way.

#### 3.2. Whole-body dynamics control

To realize the trajectory tracking of gait templates on biped robots, the WBC method is adopted as a part of the template. Next, taking robot BHR-B2 as an example, this process is described in detail.

The BHR-B2 robot is a bipedal humanoid robot developed by the Institute of Intelligent Robots of Beijing Institute of Technology, as shown in Fig. 3(a). This robot is designed to be able to move dynamically in complex environments, but it is different from other biped robots with lightweight legs [33]. The weight distribution of this robot mirrors the proportion of each limb of the human body. BHR-B2 has two legs with four joints, two arms with three joints, and 14 driving DoFs. The total weight of the robot is 40.3 kg. The robot has no roll angle of the ankle to drive the joint, so it is underactuated after contacting the ground. According to the mass ratio of the human thigh and lower leg, we attempted

to increase the height of the leg mass. Therefore, an integrated design is adopted in the thigh structure. As shown in Fig. 3(c), a lightweight thigh structure satisfying the strength requirements is designed through topology optimization. Each joint consists of a DC brush-less motor, planetary reducer, and encoder assembled in the rear of the motor. The planetary reducer has a small gear ratio of 17.43 to make direct torque control more accurate and reduce friction influence.

3.2.1. Dynamic model of robot

To verify the algorithm in this study, as shown in Fig. 3(b), we simplify the robot upper body into a rigid body with a local coordinate system  $\sum_b$ . There are six virtual DoFs between  $\sum_b$  and the inertial coordinate system  $\sum_0$ , which are three displacements  $[\varepsilon_x \ \varepsilon_y \ \varepsilon_z]^T$  and three rotations  $[\theta_x \ \theta_y \ \theta_z]^T$ . The subscripts  $r$  and  $l$  represent the right leg and left leg, respectively, and  $h, f, t$ , and  $a$  represent the joints of the hip, femur, tibia and ankle, respectively. The generalized coordinate of a biped robot floating base dynamic model  $q = [q_b^T \ q_r^T \ q_l^T]^T$  is set

$$\begin{aligned} q_b &= [\varepsilon_x \ \varepsilon_y \ \varepsilon_z \ \theta_x \ \theta_y \ \theta_z]^T \\ q_r &= [\theta_{rh} \ \theta_{rf} \ \theta_{rt} \ \theta_{ra}]^T \\ q_l &= [\theta_{lh} \ \theta_{lf} \ \theta_{lt} \ \theta_{la}]^T \end{aligned} \tag{16}$$

Through the Lagrange equation, we can obtain the dynamic equation as follows:

$$M(q)\ddot{q} + N(q, \dot{q}) = B\tau + J_r^T \lambda_r + J_l^T \lambda_l \tag{17}$$

where  $M(q) \in \mathbb{R}^{14 \times 14}$  is the inertia matrix and  $N(q, \dot{q}) \in \mathbb{R}^{14 \times 1}$  is the Coriolis force and gravity vector.  $B \in \mathbb{R}^{14 \times 8}$  is the driving torque mapping matrix and  $\tau \in \mathbb{R}^{8 \times 1}$  is the driving torque of the leg joints.  $J_r \in \mathbb{R}^{6 \times 14}$  and  $J_l \in \mathbb{R}^{6 \times 14}$  are the Jacobian matrix of the relative inertial system of the right and the left ankle, respectively, and  $\lambda_r \in \mathbb{R}^{6 \times 1}$  and  $\lambda_l \in \mathbb{R}^{6 \times 1}$  are external forces and moments on the ankle.  $\dot{q}$  and  $\ddot{q}$  are the first-order differential and the second-order differential of  $q$ , respectively.

3.2.2. Quadratic optimization formulation

The acceleration  $\ddot{q}$ , driving torque  $\tau$ , and external force of both feet are selected as optimization variables; the cost function is mainly composed of the following four parts. The acceleration at the end of the support leg is zero

$$\Gamma_1 = \|J_s^v \dot{q} + J_s^v \ddot{q}\|_{w_1}^2 \tag{18}$$

where  $\|\cdot\|^2$  is a vector two norm operator,  $w_1$  is a weight matrix with all positive numbers, and the superscript  $v$  of the Jacobian matrix represents the linear velocity component of three rows. The following cost functions are also expressed in this way. Following the desired acceleration value  $\ddot{p}_{sw}^d$  at the end of the swing leg requires

$$\Gamma_2 = \|J_{sw}^v \dot{q} + J_{sw}^v \ddot{q} - \ddot{p}_{sw}^d\|_{w_2}^2 \tag{19}$$

$$\ddot{p}_{sw}^d = k_p (p_{sw}^d - p_{sw}) + k_{\dot{p}} (\dot{p}_{sw}^d - \dot{p}_{sw}) \tag{20}$$

where  $k_p$  and  $k_{\dot{p}}$  are proportional-derivative (PD) controller parameters.  $w_2$  is a weight matrix with all positive numbers.

All trajectories are established relative to the upper body coordinate system in the trajectory generation of the gait template, so we want the upper body to remain vertical in control. Spring damping virtual model control is introduced to the roll and pitch DoF of the upper body to produce the desired reference acceleration

$$\begin{aligned} \ddot{\theta}_x^d &= -k_{roll}^1 \theta_x - k_{roll}^2 \dot{\theta}_x \\ \ddot{\theta}_y^d &= -k_{pitch}^1 \theta_y - k_{pitch}^2 \dot{\theta}_y \end{aligned} \tag{21}$$



where  $k^1$  and  $k^2$  are the stiffness and damping coefficients, respectively. Since BHR-B2 has no relevant DoFs for yaw angle control, we ignore the  $\ddot{\theta}_z^d$ . All leg joints can use a PD controller to generate reference acceleration through the reference trajectory and actual state, but for the supporting leg, we do not track the trajectory of the hip and ankle, so the optimization problem is more sparse. The tracking objective function of joint acceleration is

$$\Gamma_3 = \|B_{\ddot{q}}\ddot{q} - \ddot{q}^d\|_{w_3}^2 \tag{22}$$

where  $B_{\ddot{q}}$  is the acceleration selection matrix of the controlled DoFs.  $w_3$  is a weight matrix with all positive numbers. The reference acceleration is

$$\ddot{q}^d = [\ddot{\theta}_x^d \quad \ddot{\theta}_y^d \quad \ddot{\theta}_{st} \quad \ddot{\theta}_{swl} \quad \ddot{\theta}_{swf} \quad \ddot{\theta}_{swt} \quad \ddot{\theta}_{swa}]^T \tag{23}$$

In addition, to avoid high-frequency oscillation and sudden changes in joint torque, the objective function of joint torque is also needed

$$\Gamma_4 = \|\tau - \tau^{\text{last}}\|_{w_4}^2 \tag{24}$$

where  $\tau^{\text{last}}$  is the last driving torque of the leg joints. The objective function of the final quadratic optimization (QP) is

$$\min_{\ddot{q}, \tau, \lambda_r, \lambda_l} \Gamma_1 + \Gamma_2 + \Gamma_3 + \Gamma_4 \tag{25}$$

In addition to the constraint of the system dynamic equation of Eq. (17), joint driving capacity, and the friction cone of contact force of the supporting leg, the ZMP of the sole-plate should be within the support range (within the sole of the support foot)

$$\begin{aligned} \tilde{\lambda}_s &= \begin{bmatrix} R_s^{\text{foot}} & \mathbf{0}^{3 \times 3} \\ \mathbf{0}^{3 \times 3} & R_s^{\text{foot}} \end{bmatrix} \lambda_s \\ -L_{\text{heel}} &\leq \frac{-\tilde{\lambda}_s^{\tau_y} - \tilde{\lambda}_s^{f_x} d_{\text{an}}}{\tilde{\lambda}_s^{f_z}} \leq L_{\text{toe}} \end{aligned} \tag{26}$$

where  $R_s^{\text{foot}}$  is the rotation matrix of the supporting foot relative to the inertial system, and  $\tilde{\lambda}_s$  is the external force in the local coordinate system of the sole of the supporting foot.  $d_{\text{an}}$ ,  $L_{\text{heel}}$ , and  $L_{\text{toe}}$  are the ankle height, horizontal distance from the rear heel to the ankle, and horizontal distance from the front toe to the ankle, respectively. The superscripts of  $\tilde{\lambda}_s$  are the components of  $\tau_y$ ,  $f_x$ , and  $f_z$ , respectively. Since BHR-B2 has no roll joint of the ankle, only ZMP in the  $x$  direction is constrained here.

### 3.3. Joint level torque control

BHR-B2 uses the joint torque control mode to execute the gait trajectory. In addition to the output torque  $\tau$  controlled by the whole-body dynamics, each leg joint needs to be closed loop for the position and speed of the joint end. Therefore, the final form of control torque is

$$\begin{aligned} \tau_{ij}^c &= \tau_{ij} + k_{ij}^p (\theta_{ij}^d - \theta_{ij}) + k_{ij}^d (\dot{\theta}_{ij}^d - \dot{\theta}_{ij}) \\ i &\in \{r, l\}, j \in \{h, f, t, a\} \end{aligned} \tag{27}$$

where  $k^p$  and  $k^d$  are the PD controller gain coefficients.

## 4. Simulation and experimental results

The method presented in this paper is verified by simulation in the CoppeliaSim<sup>1</sup> simulation environment with Bullet 2.83 and by experiment with the BHR-B2 robot. We use the eigen-quadprog<sup>2</sup> QP solver

<sup>1</sup>www.coppeliarobotics.com

<sup>2</sup>https://github.com/jrl-umi3218/eigen-quadprog.git

**Table I** Dynamic parameters of robot.

Name	Mass (kg)	Length (m)	Inertia (x) <sup>†</sup>	Inertia (y) <sup>†</sup>	Inertia (z) <sup>†</sup>
Torso	29.3	0.6	1.131	0.536	0.684
Hip	1.046	0.175*	0.0008	0.001	0.0009
Femur	2.035	0.35	0.015	0.015	0.002
Tibia	1.583	0.35	0.0123	0.0128	0.0009
Foot	0.635	0.19	0.0001	0.0007	0.0008

<sup>†</sup>The moment of inertia about the coordinate axis in kg·m<sup>2</sup>.

\*The width of the hip.



**Figure 4.** The leg geometry of the robot. The inverse kinematics solution is only for the four driving DoFs of the legs.

to optimize the solution. The BHR-B2 model parameters used are shown in Table I. The initial gait template parameters in the simulation are  $T = 0.4 \sim s$ ,  $\alpha = 0.5$ ,  $v_x^d = 0 \sim m/s$ ,  $v_y^d = 0 \sim m/s$ ,  $H_s^d = 0.7 \sim m$ ,  $h_{sw}^d = 0.1 \sim m$ , and  $W_s^d = 0.0375 \sim m$ . Some parameters in the control are as follows:  $d_{an} = 0.0645 \sim m$ ,  $L_{toe} = 0.12 \sim m$ , and  $L_{heel} = 0.07 \sim m$ . The gain parameters in this method, PD controller, and weight coefficient in optimization can be set manually and by conventional experiments, which are not shown here.

#### 4.1. Inverse kinematics of BHR-B2

The robot increases the initial position between the two feet through the paranoia of the y-direction of the legs to increase the robot's adjustment range to deal with the changes in the complex environment (Fig. 4). In addition, the reference track will change suddenly when the leg state of the state template changes. Therefore, the inverse kinematic algorithm based on the QP solution is used to obtain a smooth trajectory similar to [34].

According to the setting of the gait template, we need to solve the pose of each leg end (the foot) relative to the upper body. We set the endpoint as the front edge of the sole to include the ankle joint, which is conducive to constraining the pose of the sole relative to the ground. We choose the increment of the joint angle  $\Delta q_{r,l} \in \mathbb{R}^4$  as the optimization variables to meet the joint angle limit more conveniently; the lower corner mark indicates the right or left leg. Finally, the solution results are obtained according

to the current leg state  $q_{r,l} \in \mathbb{R}^4$ . The optimization objective consists of three parts. The first part is the control of the end position and speed

$$\Psi_1 = \left\| \hat{J}_{r,l} \Delta q_{r,l} - \hat{v}_{r,l}^d \Delta T \right\|_{Q_1}^2$$

$$\hat{v}_{r,l}^d = v_{r,l}^d + k_{IK} (p_{r,l}^d - p_{r,l}) \tag{28}$$

where  $\hat{J}_{r,l}$  represents the Jacobian matrix of the leg,  $\hat{v}_{r,l}^d$  represents the target speed at the end of the foot, which is obtained by superimposing the position error on the reference speed  $v_{r,l}^d$ ,  $p_{r,l}$  represents the current position, and  $p_{r,l}^d$  represents the reference position.  $k_{IK}$  is a constant gain parameter, which we set to 10. The control period  $\Delta T$  is 1~ms in the simulation and experiment.  $Q_1$  is a weight matrix with all positive numbers.

The second part is to keep the increment as small as possible. In addition, to avoid the oscillation of the solution, the constraint with the last result  $\Delta q_{r,l}^{\text{last}}$  should be added

$$\Psi_2 = \left\| \Delta q_{r,l} \right\|_{Q_2}^2 + \left\| \Delta q_{r,l} - \Delta q_{r,l}^{\text{last}} \right\|_{Q_3}^2 \tag{29}$$

$Q_2$  and  $Q_3$  are weight matrices with all positive numbers.

The third part is the optimal solution for the foot posture. Because the joint configuration of the robot's legs cannot control the position and posture of the foot at the same time, only the control of the foot pitch angle remains

$$\Psi_3 = \left\| \zeta(\Delta q_{r,l}) - \theta_{(r,l)a} \right\|_{Q_4}^2 \tag{30}$$

where  $\zeta(\Delta q_{r,l})$  represents the forward kinematics of the pitch angle of the foot relative to the upper body calculated from the joint angle and  $\theta_{(r,l)a}$  represents the desired pitch angle of the footplate. Because the torque control of the robot's ankle joint will make it conform to the ground and produce different angles, when it becomes a swinging leg,  $\theta_{(r,l)a}$  needs to interpolate from the current angle to the desired angle. This process can solve the sudden change of inverse kinematics when changing the leg state. The objective function of the final QP optimization is

$$\min_{\Delta q_{r,l}} \Psi_1 + \Psi_2 + \Psi_3 \tag{31}$$

The constraints of the optimization problem include joint velocity and angle constraints:

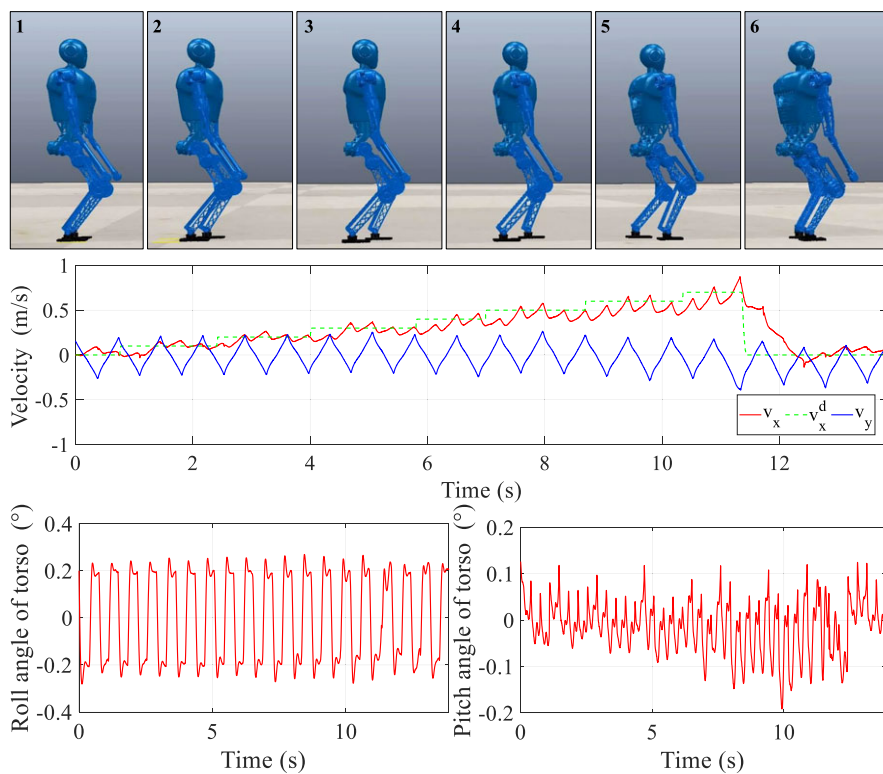
$$\phi_{\min} \leq q_{r,l} + \Delta q_{r,l} \leq \phi_{\max}$$

$$\dot{\phi}_{\min} \leq \frac{\Delta q_{r,l}}{\Delta T} \leq \dot{\phi}_{\max} \tag{32}$$

where  $\phi_{\max}$ ,  $\phi_{\min}$ ,  $\dot{\phi}_{\max}$ , and  $\dot{\phi}_{\min}$  represent the maximum and minimum angle and speed limits. For example, the knee joint angle of the leg cannot be positive to avoid anti-joint effects. The final solution of inverse kinematics is  $q_{r,l}^d = q_{r,l} + \Delta q_{r,l}$ .

#### 4.2. Dynamically adjusted template parameters in simulation

First, the dynamic variable speed is tested in the simulation. By randomly changing the reference speed  $v_x^d$  of the gait template with  $v_y^d = 0$ , the effectiveness of real-time planning of the template and robustness of the control algorithm are verified, as shown in Fig. 5. We gradually increase the reference speed to a maximum of 0.7 m/s in steps of 0.1 m/s. With the increase in  $v_x^d$ , the step length of the robot changes significantly through the screenshots, which shows the real-time effectiveness of the gait template. The actual speed  $v_x$  stably tracks the desired speed with small error. The lateral velocity fluctuates periodically around zero, because the ankle lacks the DoF in this direction. The speed tracking effect indicates that the speed control in the controller is adequate. The posture angle of the upper body changes in a small range during variable speed walking, showing the control algorithm's effectiveness. The upper body should be kept upright to increase the stability of walking.



**Figure 5.** Walking simulation under different reference speeds. Above: screenshots of the simulation experiment. Central: running speed curve of the robot. Below: roll and pitch angle of the robot's torso posture.

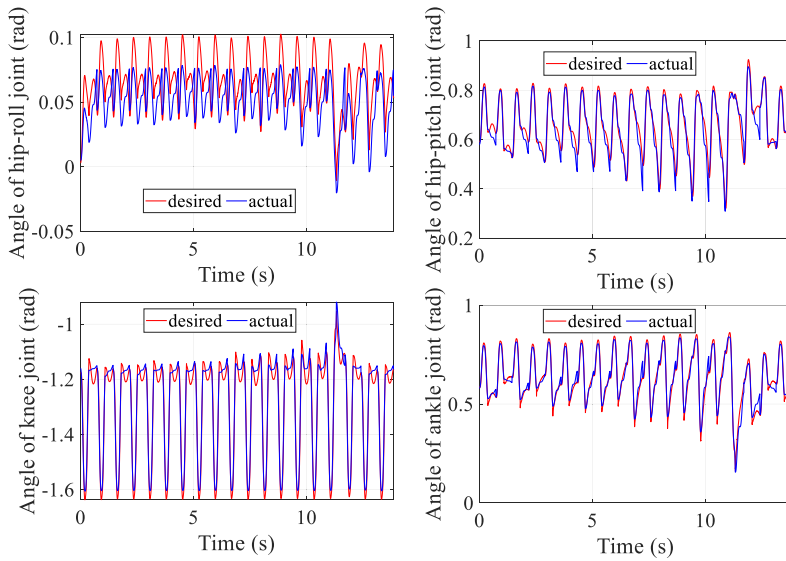
Figure 6 shows the tracking of the driving joint of the right leg during walking, and the gait shows periodicity. The trajectory keeps the robot in dynamic balance, which further illustrates the real-time effectiveness of the gait template.

The simulation experiment of dynamic height change is shown in Fig. 7. Stepped reference height changes are smoothed by gait templates, enabling the robot to maintain balance. The stable convergence after the speed mutation indicates the effectiveness of the adjustment in the gait template. Thus, dynamic and accurate height control is realized through a gait template and control algorithm.

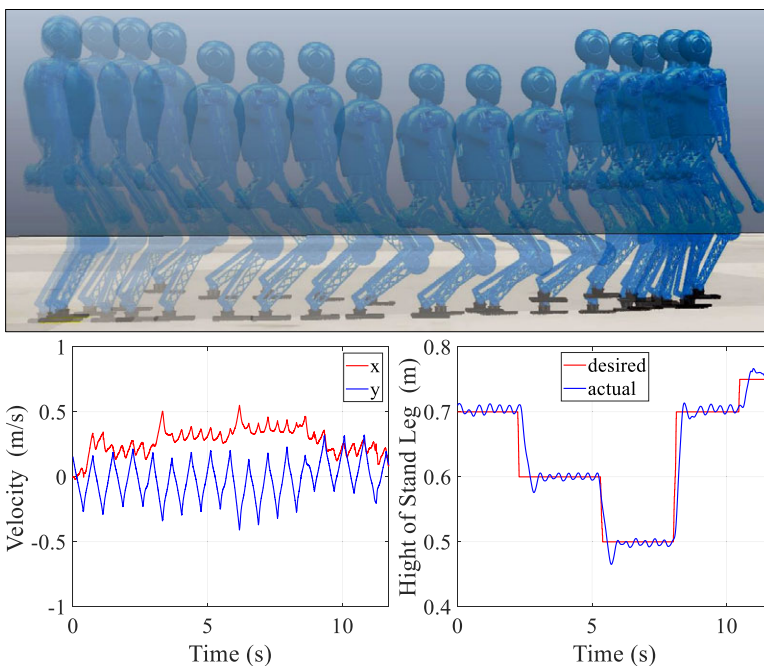
### 4.3. Steady motion experiment

The verification algorithm on a real robot is different from the simulation system because it is difficult to obtain the real value of the robot state and there is considerable noise in the sensor. Friction and other factors produce more inaccuracies in direct drive joint torque control (no torque feedback at the joint end). In the experiment, the state estimation is completed by the Kalman filter (KF) method. The velocity of the joint is obtained by numerical differentiation of the position and filtered by a first-order KF. The robot determines whether it touches the ground by force sensors installed on the soles of its feet. The communication mode of EtherCAT enables the whole control to be completed at a frequency of 1 kHz.

The stable motion control has been tested on the physical robot BHR-B2, as shown in Fig. 8. The controller makes the robot move at a predetermined speed, and the angle error of the upper body is controlled within  $1^\circ$ . The shaking of the upper body will cause some interference to the speed control, but the controller can stabilize the robot within this range.

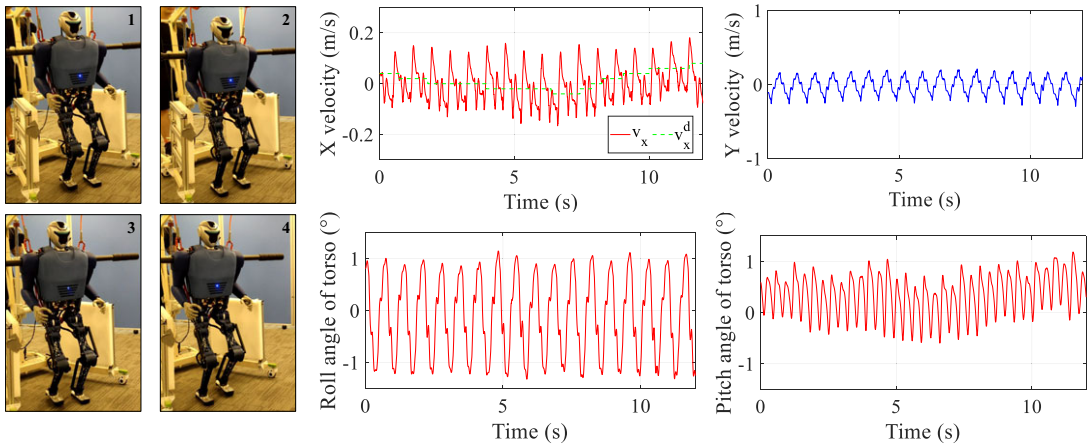


**Figure 6.** The tracking curve of four joint angles for right leg in the simulation experiment of dynamic speed change. The red curve represents the desired state and the blue curve represents the actual state.

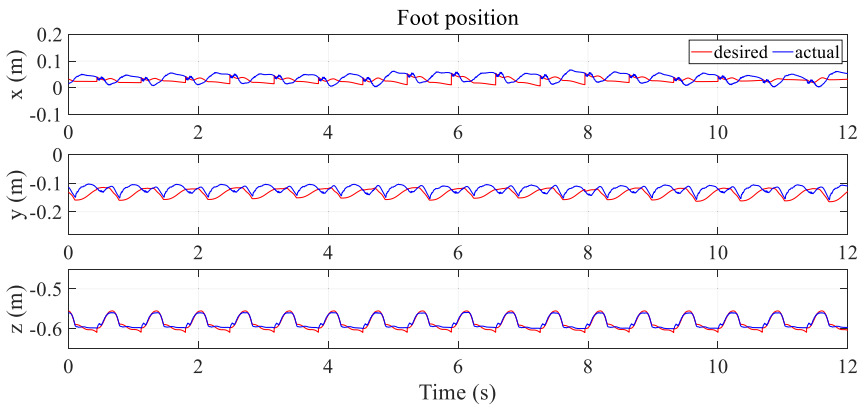


**Figure 7.** Simulation experiment of a gait template with variable support height. Screenshot of robot simulation and tracking curve of speed and support height. When the robot moves from left to right, the reference height decreases and then increases.

Figure 9 shows the spatial position track of the foot end relative to the upper body generated by the gait template and the actual tracking. Due to the influence of joint friction, the trajectory tracking is not ideal when the motion range of the end of the leg is small in the  $x$  and  $y$  directions. Especially in the leg support stage, considerable position tracking accuracy is lost to maintain the upper body posture.



**Figure 8.** BHR-B2 stable motion experiment. The protective rope is loose and does not give any support to the robot. The carbon fiber tubes on both sides of the robot’s shoulders are added for protection.

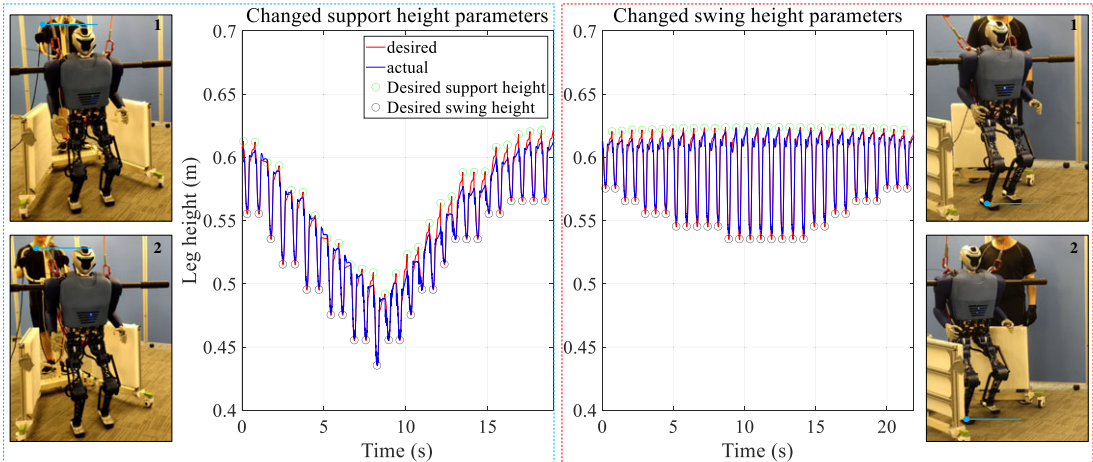


**Figure 9.** Gait template desired trajectory and actual tracking effect during steady motion. The curve represents the position of the leg end relative to the upper body, so the z-direction is negative. When the value in the z-direction is smooth, it is the support stage of the current leg.

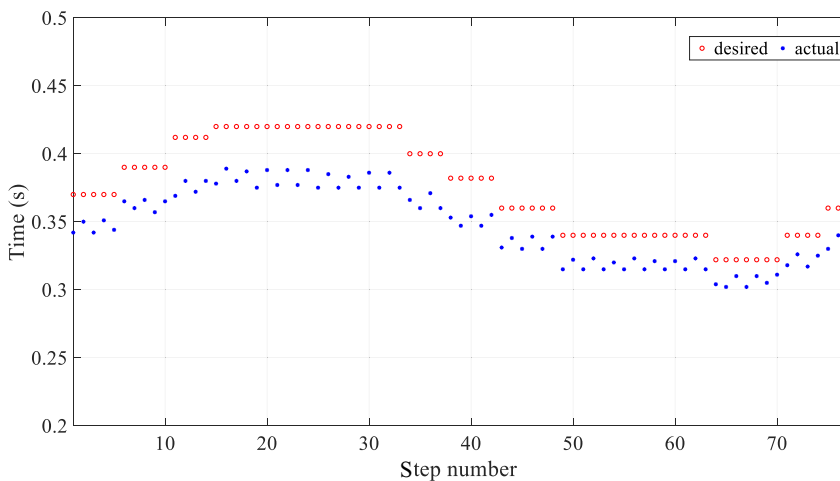
However, the position control in the z-direction shows good tracking performance without the influence of other control targets. When the leg changes from the support phase to the swing phase, the reference position will change suddenly to the current actual position to avoid the sudden change in control torque, which is also the trajectory switching effect caused by the event trigger in the gait template. Using the method in subsection 2.2, the gait template can effectively transition this process.

**4.4. Dynamic variable template parameter in the experiment**

We also experimented with dynamically adjusting the template parameters when the robot changes the height of the supporting leg, the height of the lifting foot, and the walking cycle. Figure 10 shows the experimental results of the first two parameters. The support height changes in steps from 0.48 to 0.62 m, and the curve represents the periodic position change of the upper body relative to the end of the right leg. The maximum value represents the expected support height, and the relative value in the process of reciprocating changes represents the height of leg lifting. The foot lifting height changes from 0.05 to 0.1 m. The curve represents the periodic position change of the upper body relative to the end of the right leg. The maximum value represents the expected support height, and the relative value in the



**Figure 10.** Dynamic adjustment of support height and swing foot lifting height. The left side is the variable height experiment, and the right side is the variable foot lifting height experiment. The numerical interval between the green circle and the black circle indicates the height of foot lifting. The blue line in the video screenshot is marked with a reference datum to make this change more obvious.



**Figure 11.** Variable walking cycle time experiment. The horizontal axis represents the number of steps. The red circle indicates the expected walking cycle time, and the blue dot indicates the actual running time.

process of reciprocating changes represents the height of leg lifting. Therefore, the experiment verifies the dynamic correctable characteristics of the gait template, and the control algorithm enables the robot to maintain balance.

Figure 11 shows the change in the walking cycle. The slight difference between the reference walking cycle and the actual walking cycle is because the swinging leg touches the ground in advance, which makes the leg state change. The real-time planning of the gait template causes smooth switching between two gait frequencies, and the robot can move steadily. There are differences between the two adjacent blue dots in the figure, which indicates that the performance of the left and right legs on the robot is not the same, which again indicates the challenge of experimental verification.

## 5. Conclusions

This study uses the UBR without roll DoF in the ankle to verify the algorithm. The simulation and experimental results show the feasibility of the gait template and the algorithm's robustness. In real time, we can generate a walking gait with variable speed, step size, and metamorphic center height through the proposed model-free walking template. This advantage makes the real-time adjustment more reliable and faster. The template parameters which can be changed in real time will grant the robot improved robustness in the dynamic environment. For example, the robot can squat through low spaces, change the lifting height in rugged terrain, and accelerate the gait frequency (reduce the walking cycle) after being significantly disturbed to achieve rapid stability. Flexible motion generation and stable control methods will contribute to the further development of human–robot cooperation and interaction, such as remote synchronization of human and robot actions [35]. In this study, we have tested the feasibility of a gait template with dynamically adjusted parameters and the effectiveness of the control algorithm. Further verification in complex environment application scenarios has not been carried out, which will be the focus and direction of our future work. It should be noted that there are still some limitations in verifying this algorithm in complex environments, such as the real-time and accuracy requirements of environmental awareness and parameter decision algorithms in different scenarios.

In future work, we will use multi-parameter optimization for the end planning of the swing leg to adapt to more complex environments and mission scenarios, such as climbing stairs. We believe that the artificial intelligence algorithm may help in dealing with the multi-parameter adjustment in complex situations [30]; we will also try to use neural networks to adjust gait template parameters cooperatively.

**Acknowledgments.** This work was supported by the National Key Research and Development Program under Grant (2018YFE0126200) and the National Nature Science Foundation of China under Grant (62103053). We thank Wei-wei Wang and Min Zhu for their contributions to the construction of the robot system.

**Author's contributions.** Lianqiang Han put forward the theory and carried out experimental verification; Xuechao Chen and Zhangguo Yu improved and revised the idea and manuscript; Zhifa Gao and Jintao Zhang designed the experimental process; Gao Huang designed the platform; Kenji Hashimoto and Qiang Huang supervised and managed the project. All authors have read and agreed to the published version of the manuscript.

**Competing interests declaration.** The authors declare none.

**Ethical considerations.** None.

## References

- [1] A. Namiki and S. Yokosawa, "Origami folding by multifingered hands with motion primitives," *Cyborg Bionic Syst.* **2021**, 1–15 (2021).
- [2] X. Chen, Z. Yu, W. Zhang, Y. Zheng, Q. Huang and A. Ming, "Bioinspired control of walking with toe-off, heel-strike, and disturbance rejection for a biped robot," *IEEE Trans. Ind. Electron.* **64**(10), 7962–7971 (2017).
- [3] K. Huang, Y. Xian, S. Zhen and HSun, "Robust control design for a planar humanoid robot arm with high strength composite gear and experimental validation," *Mech. Syst. Signal. Process.* **155**, 107442 (2021).
- [4] C. Zeng, H. Su, Y. Li, J. Guo and C. Yang, "An approach for robotic leaning inspired by biomimetic adaptive control," *IEEE Trans. Industr. Inform.* **18**(3), 1479–1488 (2021).
- [5] T. Fukuda, "Cyborg and bionic systems: Signposting the future," *Cyborg Bionic Syst.* **2020**, 1–2 (2020).
- [6] Boston Dynamics, "ATLAS", Available at: [www.bostondynamics.com/atlas](http://www.bostondynamics.com/atlas), accessed on 15 June 2022.
- [7] C. Liu, J. Yang, K. An, M. Liu and Q. Chen, "Robust control of semi-passive biped dynamic locomotion based on a discrete control lyapunov function," *Robotica* **38**(8), 1345–1358 (2020).
- [8] D. Kim, Y. Zhao, G. Thomas, B. R. Fernandez and L. Sentis, "Stabilizing series-elastic point-foot bipeds using whole-body operational space control," *IEEE Trans. Robot.* **32**(6), 1362–1379 (2016).
- [9] J. Siekmann, K. Green and J. Warila, "Blind bipedal stair traversal via sim-to-real reinforcement learning," arXiv preprint arXiv: 2105.08328, 2021.
- [10] Agility Robotics, "Digit tackles wet and muddy hills", Available at: [www.youtube.com/watch?v=bV3KnthEY2c](https://www.youtube.com/watch?v=bV3KnthEY2c), accessed on 15 June 2022.



- [11] X. Da, R. Hartley and J. W. Grizzle, “Supervised Learning for Stabilizing Underactuated Bipedal Robot Locomotion, with Outdoor Experiments on the Wave Field,” **In: IEEE International Conference on Robotics and Automation** IEEE (2017) pp. 3476–3483.
- [12] S. Kajita, F. Kanehiro and K. Kaneko, “Biped Walking Pattern Generation by Using Preview Control of Zero-Moment Point,” **In: IEEE International Conference on Robotics and Automation** IEEE (2003) pp. 1620–1626.
- [13] J. Luo, Y. Su, L. Ruan, Y. Zhao, D. Kim, L. Sentis and C. L. Fu, “Robust bipedal locomotion based on a hierarchical control structure,” *Robotica* **37**(10), 1750–1767 (2019).
- [14] G. Mesesan, J. Engelsberger and G. Garofalo, “Dynamic Walking on Compliant and Uneven Terrain Using DCM and Passivity-Based Whole-Body Control,” **In: IEEE-RAS 19th International Conference on Humanoid Robots** IEEE (2019) pp. 25–32.
- [15] M. Pi, Y. Kang, C. Xu, G. Li and Z. J. Li, “Adaptive time-delay balance control of biped robots,” *IEEE Trans. Ind. Electron.* **67**(4), 2936–2944 (2019).
- [16] X. Xiong and A. Ames, “Orbit Characterization, Stabilization and Composition on 3D Underactuated Bipedal Walking via Hybrid Passive Linear Inverted Pendulum Model,” **In: IEEE/RSJ International Conference on Intelligent Robots and Systems** IEEE (2019) pp. 4644–4651.
- [17] X. Xiong and A. Ames, “SLIP walking over rough terrain via H-LIP stepping and backstepping-barrier function inspired quadratic program,” *IEEE Robot. Autom. Lett.* **6**(2), 2122–2129 (2021).
- [18] B. Dadashzadeh and C. J. Macnab, “Slip-based control of bipedal walking based on two-level control strategy,” *Robotica* **38**(8), 1434–1449 (2020).
- [19] A. Hereid, O. Harib, R. Hartley, Y. Gong and J. W. Grizzle, “Rapid Trajectory Optimization Using c-Frost with Illustration on a Cassie-Series Dynamic Walking Biped,” **In: IEEE/RSJ International Conference on Intelligent Robots and Systems** IEEE (2019) pp. 4722–4729.
- [20] M. Hobon, V. De-León-Gómez, G. Abba, Y. Aoustin and C. Chevallereau, “Feasible speeds for two optimal periodic walking gaits of a planar biped robot,” *Robotica* **40**(2), 377–402 (2022).
- [21] L. Drnach and Y. Zhao, “Robust trajectory optimization over uncertain terrain with stochastic complementarity,” *IEEE Robot. Autom. Lett.* **6**(2), 1168–1175 (2021).
- [22] K. Scott, D. Robin and F. Maurice, “Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot,” *Auton. Robot.* **40**(3), 429–455 (2016).
- [23] J. W. Grizzle, C. Christine and C. L. Shih, “HZD-Based Control of a Five-Link Underactuated 3D Bipedal Robot,” **In: IEEE Conference on Decision and Control** IEEE (2008) pp. 5206–5213.
- [24] M. H. Raibert. *Legged Robots That Balance* (MIT Press, USA, 1986).
- [25] G. Nelson, A. Saunders and R. Playter, “The PETMAN and Atlas Robots at Boston Dynamics,” **In: Humanoid Robotics: A Reference**, Goswami A. and Vadakkepat P., Springer, Dordrecht, (2019) pp. 169–186.
- [26] L. Han, X. Chen, Z. Yu, X. Zhu, K. Hashimoto and Q. Huang, “Trajectory-free dynamic locomotion using key trend states for biped robots with point feet,” *Sci. China Inf. Sci.*, (2022). doi: [10.1007/s11432-021-3450-5](https://doi.org/10.1007/s11432-021-3450-5).
- [27] L. Han, X. Chen, Z. Yu, Z. Gao, Y. Huang and Q. Huang, “Balance control of underactuated biped robot for discrete terrain,” *Acta Autom. Sin.* **48**(9), 1–11 (2022).
- [28] K. Yin, L. Kevin and V. Michiel, “Simbicon: Simple biped locomotion control,” *ACM Trans. Graph.* **26**(3), 105 (2007).
- [29] W. Qi and H. Su, “A cybertwin based multimodal network for ECG patterns monitoring using deep learning,” *IEEE Trans. Ind. Inform.* **18**(10), 6663–6670 (2022).
- [30] H. Su, Y. Hu, H. R. Karimi, A. Knoll, G. Ferrigno and E. De Momi, “Improved recurrent neural network-based manipulator control with remote center of motion constraints: Experimental results,” *Neural Netw.* **131**, 291–299 (2020).
- [31] Z. Li, X. Cheng, X. Peng, P. Abbeel, S. Levine, G. Berseth and K. Sreenath, “Reinforcement Learning for Robust Parameterized Locomotion Control of Bipedal Robots,” **In: IEEE International Conference on Robotics and Automation (ICRA)** IEEE (2021) pp. 2811–2817.
- [32] L. Krishna, G. A. Castillo, U. A. Mishra, A. Hereid and S. Kolathaya, “Linear policies are sufficient to realize robust bipedal walking on challenging terrains,” *IEEE Robot. Autom. Lett.* **7**(2), 2047–2054 (2022).
- [33] X. Zhu, L. Wang, Z. Yu, X. Chen and L. Han, “Motion Control for Underactuated Robots Adaptable to Uneven Terrain by Decomposing Body Balance and Velocity Tracking,” **In: IEEE International Conference on Advanced Robotics and Mechatronics (ICARM)** IEEE (2021) pp. 729–734.
- [34] S. Y. Fengy, E. Whitmanz, X. Xinjilefuy and C. G. Atkeson, “Optimization Based Full Body Control for the Atlas Robot,” **In: IEEE-RAS International Conference on Humanoid Robots (Humanoids)** IEEE (2014) pp. 120–127.
- [35] H. Su, W. Qi, Y. Schmirander, S. Ovrur, S. Cai and X. Xiong, “A human activity-aware shared control solution for medical human-robot interaction,” *Assembly Autom.* **42**(3), 388–394 (2022).