

Research Article

Real-Time Footprint Planning and Model Predictive Control Based Method for Stable Biped Walking

Song Wang ¹, Songhao Piao ¹, Xiaokun Leng,¹ Zhicheng He,¹ Xuelin Bai,²
and Li Huazhong³

¹School of Computer Science, Harbin Institute of Technology, Harbin, China

²Leju Robotics, Shenzhen, China

³Shenzhen Institute of Information Technology, Shenzhen, China

Correspondence should be addressed to Song Wang; song.w@hit.edu.cn

Received 11 January 2022; Revised 27 January 2022; Accepted 11 February 2022; Published 1 April 2022

Academic Editor: Akshi Kumar

Copyright © 2022 Song Wang et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In order to walk in a physical environment, the biped will encounter various external disturbances, and walking under persistent conditions is still challenging. This paper tries to improve the push recovery performance based on capture point (CP) and model predictive control. The trajectory of zero moment point (ZMP) and center of mass are solved and predicted in a limited time horizon. Online footprint generator is combined with MPC walking pattern generation, which can keep biped stable in the next few steps, and projection of ZMP is used to calculate the next footprint and reach the target CP in an incremental way. Verification of the proposed stable biped walking method is conducted by simulation and experiments.

1. Introduction

tBiped robot is one kind of mobile robot and belongs to the field of leg robot. In the world of robotics, biped walking is one of the fields with the most interesting research [1]. Robots need to interact and collide intermittently with the environment to complete specific actions and tasks [2]. Compared with other forms of robots, biped robot has more advantages: the biped with a humanoid structure can adapt to various environments, and it can replace humans in many dangerous jobs and offer services for humans, such as artificial limb and medical rehabilitation [3].

The biped robot model is nonlinear dynamic with multiple degrees of freedom and constraints, and it is important to interpret and predict the long-term behavioral trends of the dynamic system for various control tasks. A simplified biped walking linear inverted pendulum model (LIPM) is the baseline of humanoid walking pattern controller [4, 5], and passive walking and under-actuated robots have certain advantages in energy-saving walking. But passive walking has poor terrain adaptability and can only

walk downhill or on flat ground; while under-actuated walking robots require special structural design and rely on complex walking control strategies [6]. Zhang et al. [7] proposed a walking pattern generator for omnidirectional walking on a slope and uneven plane and realized the oblique walk and heading for any direction, but the performance of push recovery under external persistent force is not analyzed. Oliver et al. proposed an improved model named the flexible linear inverted pendulum model (FLIPM) which integrate damped spring and second center of mass to LIP [8], which realized a stable walk on a servo-driven biped robot. However, the performance is only improved in the z direction, which is not included by FLIPM and causes a rotation of the root alongside the x -axis towards the ankle joint lifting up from the plane. Biped robot has dozens of degrees of freedom and no fixed base, which leads to its dynamic system is very complex. Juang et al. proposed the fully connected recurrent neural network optimized by continuous multiobjective ant colony optimization to form CpG to solve the multiobjective gait generation problem of Nao robot. An open-source multiple DOF servo-driven

robot Robotis op [9] is used to test their method performance, but the coupled oscillator model is not able to keep stability under an unpredictable perturbation.

In the process of walking, the biped robot will inevitably be affected by various uncertain factors, such as the uncertainty of mathematical model based on different assumptions [10], the uncertainty of robot system parameters caused by the structural size, material properties, manufacturing and assembly errors, and the random uncertainty of driving torque caused by actuator noise and joint friction, and the external environment disturbance and plane condition uncertainty in the process of robot walking [11]. Pratt et al. [12] introduced the concepts of capture point and capture area by LIPM and flywheel model, and CP is the key metric for push recovery of the biped which shows the target direction to maintaining body stability by keeping center of pressure (CoP) inside the foot support area. But how the capture region adapts with other biped models is not discussed. Majid et al. generate walking patterns for the biped by two stages: computes the best step location and duration and adapts these values using divergent component of motion (DCM) measurement. Engelsberger [13] designed CP following and CP step-end controller and has proved the antipush stability of CP control principle, but real-time footprint position adjustment is not included to enhance the robustness of the walking pattern against external perturbations. Since the large number of joints, it is very complex to use CPG model to describe the joint angle for the 3D humanoid robot. Q-learning-based CPG model [14, 15] can keep the body recover from horizon perturbation, but it need different expertise to redesign the controller. In addition, the motion obtained by this gait control method is generally not optimal, and the anti-interference in vertical direction ability is poor.

Humanoid walking pattern generation can be solved by model predictive control (MPC) with efficient constraint handling [16]. The optimization-based method is utilized so as to generate, at each iteration, the optimal gait trajectory of the system satisfying the given constraints. However, it is well known that the action of a bounded persistent disturbance can destabilize a predictive controller which has been designed to be stabilizing for the nominal case [17]. These controllers can be used with a cost function that at the specific problem of generating trajectory by quadratic programming in linear system control, a fast optimization algorithm is proposed. However, several papers like [18–20] used restrictions on model state at the end of the prediction horizon. Since the state of the system is completely dependent on the beginning and the end of the horizon, motions lack the flexibility to perform secondary control objectives. Amos et al. [21] proposed the strategy architecture designed for end-to-end training, the robot learned to combine high-level planning strategy with low-level motion controller to realize autonomous navigation on a curved path, which is less computationally and memory intensive compared to traditional optimization solutions, but the model used in that article is still linear which is hard to extend to complex systems like the biped walking. Scianca et al. [22] introduced an IS-MPC framework for gait

generation which extend LIPM with ZMP trajectory input, and recursive feasibility of internal stability is realized for biped dynamics, but the cost of control and response speed of method are not analyzed for position control based humanoid robot [23].

The gait trajectory planning method based on LIP does not consider the external interference force, so it cannot walk in a contact-rich environment. The gait pattern generator can only calculate the trajectory position at the next time according to the state transition equation but cannot predict the trajectory in the future. This paper proposes footprint planning and MPC method for stable biped walking considering the capture point principle. The framework of the method is summarized in Figure 1. LIPM is used to linearize the dynamic model of biped walking, and MPC is used as a gait pattern generator to generate a smooth trajectory of CoM. Online footstep generator based on capture point feedback is combined with the MPC controller, and the biped can take one or more steps to reach the stable state and realize omnidirectional walking on the horizon plane. And a projection function maps the target ZMP into support polygon which makes inverse kinematics is solvable.

2. Materials and Methods

2.1. Approximation of Walking Model. Approximation acts effective method to deal with complex systems. Most humanoid robots use the model-based walking planning method to get the gait pattern and abstract the biped dynamics equation from the model by approximating the center of mass (CoM). The classical inverted pendulum model is widely used for humanoid walking pattern generation. Some conditions are mentioned first as follows:

- (1) Whole-body mass of the robot is concentrated on the CoM.
- (2) CoM height h_{CoM} in 3-dimensional space is invariant.
- (3) Biped legs are massless and contact with the ground through a pivot that can be rotated.

Under the joint action of the supporting moment and the force along the stretching direction of the rod, the dynamic equation of LIPM is established, and the trajectory of the CoM is obtained by solving the dynamic equation. Using the LIPM, let p denote the position of the center of pressure (CoP) on the plane, and the horizontal dynamics differential equation of CoM is

$$\ddot{x} = \omega_0^2 (x - p), \quad (1)$$

where $\omega_0 = \sqrt{g/h_{CoM}}$ and g is gravity force, and h_{CoM} is the constant height of CoM along z -axis. x denotes the position of CoM along x -axis, and \ddot{x} denotes body acceleration. This approximation (1) decouples the sagittal and coronal motions of the biped robot, so we will focus on the x -axis motion throughout this paper, and y direction motion is totally identical.

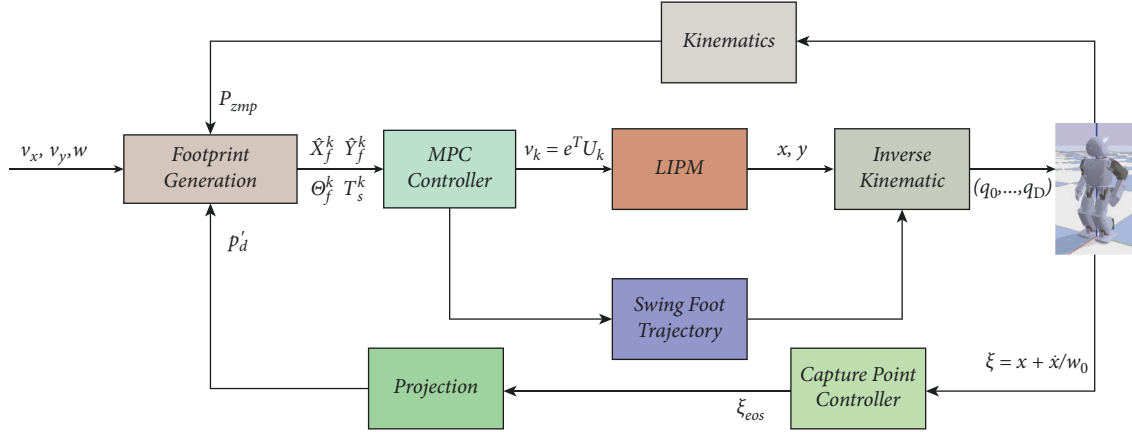


FIGURE 1: Block scheme of the proposed CP- and MPC-based framework for walking gait.

Considering the discretization at a minimum interval time T , CoM, and CoP are discretized. Let input control signal $u(t) = \dot{x}$ which is the jerk of CoM position in state-space equation, then the system state transition at time $t = kT, k = 1, 2, \dots$ with notation

$$\hat{x}_k = \begin{bmatrix} x(kT) \\ \dot{x}(kT) \\ \ddot{x}(kT) \end{bmatrix}, u_k = \dot{x}(kT), p_k = p(kT). \quad (2)$$

From (2), we can get the ZMP position

$$p_k = \begin{bmatrix} 1 & 0 & 1/w_0^2 \end{bmatrix} \hat{x}_k. \quad (3)$$

For the system state \hat{x}_k , the discretized system of LIPM is

$$\begin{aligned} \hat{x}_{k+1} &= \begin{bmatrix} 1 & T & T^2/2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \hat{x}_k + \begin{bmatrix} T^3/6 \\ T^2/2 \\ T \end{bmatrix} u_k \\ &= A\hat{x}_k + Bu_k. \end{aligned} \quad (4)$$

In the process of bipedal walking, the position of ZMP p_k always falls into the support polygon of the foot, so the range of ZMP can be limited within

$$p_k^{\min} < p_k < p_k^{\max}. \quad (5)$$

The range of ZMP $[p_k^{\min}, p_k^{\max}]$ depends on foot contact condition with the horizontal plane at time kT . According to the state transition (2), the next state at time $k+1$ is decided by the current state x_k and input control u_k . The core idea of MPC makes the actual ZMP p_k most likely close to reference ZMP trajectory p_k^{ref} at the minimum control cost at time kT . Then, the optimal control sequence u_k, u_{k+1}, \dots can be get by solving the quadratic problem (QP).

$$\min_{u_k, u_{k+1}, \dots} \sum_{i=k}^{\infty} \frac{1}{2} \alpha (p_{i+1} - p_{i+1}^{ref})^2 + \frac{1}{2} \beta u_i^2, \quad (6)$$

where $\alpha/\beta (\alpha > 0, \beta > 0)$ are factors to balance the system response ratio and control cost. If α/β rises, the tacking speed of system and control costs increase. And if α/β is lower, the tracking speed of the system and control costs decrease.

2.2. Capture Point Controller. For push resistance of the bipedal, instantaneous capture point (ICP) was utilized. During the motion of a LIP model, there exists a metric point on a walking plane, where placing the center of the landing foot can stabilize the motion of the model. This point can be calculated as

$$\xi = x + \frac{\dot{x}}{w_0}, \quad (7)$$

where ξ is the x or y component of ICP position on plane. Due to the limitation of robot dynamics, when the capture point is not captured, the CP will be away from CoP in a straight line until it falls down that is noted by Figure 2.

The features of the capture point can be described as follows:

- (1) Due to the regular dynamics of the robot, if the point is not captured, with time, the ICP moves in a straight line joining the CoP and the CP in a direction away from the foot, as shown in Figure 2.
- (2) Zero-step capturability: If the ICP falls inside the support polygon, then the robot can balance itself with the application of a good control like MPC; however, if the ICP falls outside the support polygon, the robot will have to take at least one step to protect itself from falling down.
- (3) One-step capturability: The model can come to stability within a step if the ICP falls within the reachable range. The reachable region is bounded by the maximum step length.

According to equation (1) and equation (7), the dynamic relationship between CP and ZMP is

$$\dot{\xi}_x = w_0 (\xi_x - p_x). \quad (8)$$

Since the pole of transition function is w_0 , equation (7) is an unstable system without external additional input. The idea of CP control is to generate the ZMP trajectory so that the current CP reaches the target CP within a given time interval, and then, the CoM follows the planning CP curve. The solution of equation (7) in the time domain is

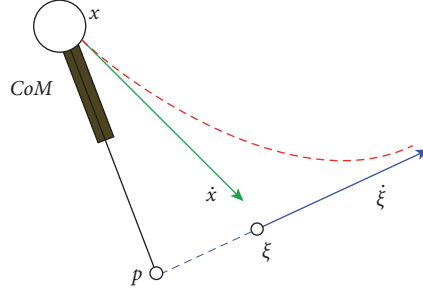


FIGURE 2: Top view of the LIPM with point foot for a given initial state.

$$\xi_x(t) = e^{\omega_0 t} (\xi_{x,0} - p_x) + p_x, \quad (9)$$

where $\xi_{x,0}$ is the initial position of CP. It can be seen that when ZMP p_x is constant, CP will change exponentially. Let $\xi_{x,d}$, ξ_x represent the target CP and current CP, respectively, and time from $\xi_{x,d}$ to ξ_x is dT , and then, (9) can be discretized as

$$p_x = \frac{\xi_{x,d} - e^{\omega_0 dT} \xi_x}{1 - e^{\omega_0 dT}} = \frac{1}{1 - e^{\omega_0 dT}} \xi_{x,d} - \frac{e^{\omega_0 dT}}{1 - e^{\omega_0 dT}} \xi_x. \quad (10)$$

The goal of end-of-step CP controller: ZMP trajectory is generated by $\xi_{x,d}$ and variable time span dT . Let $\xi_{x,eos}$ represent the end of step at the ending of each foothold, and $\xi_{x,d} = \xi_{x,eos}$. Suppose the center of the support foot is approximately regarded as the ZMP point p_i at step i , and the initial CP is

$$\xi_{init,i} = p_i + (\xi_{eos,i} - p_i) e^{\omega_0 t_{step}}, \quad (11)$$

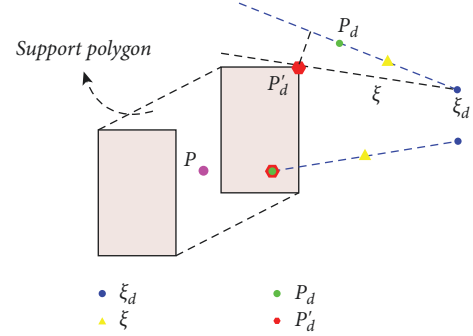
where t_{step} is the duration of each step. Iterative relation of CP state is $\xi_{eos,i-1} = \xi_{init,i}$. When the robot is subjected to external thrust, CP changes suddenly, and the biped needs to replan the landing footprint so that the CoM is adjusted to ξ_{eos} . According to the iterative relationship (11), the CP position in each support leg cycle can be calculated. Then, the system reaches a stable state following the new reference ZMP sequence generated by the CP controller.

ZMP reference trajectory is a continuous curve and constrained by robot dynamics of support polygon. When the system is disturbed by a large external force, CP varies fast out of feet support polygon on the plane. If p_x is calculated by (11), \hat{p}_d might locate out of support polygon. And the incremental progressive method can be adopted for ZMP, and CoM is controlled to move along the target CP direction within the current support polygon. Formula (9) is differentiated as follows:

$$d\xi_{eos} = \frac{\partial \xi_{eos}}{\partial p_d} = (1 - e^{\omega_0 dT}) dp_d, \quad (12)$$

$$\Delta \xi_{eos} = \int d\xi_{eos} dt.$$

A method to implement the incremental integration (12) is the ZMP projection as shown in Figure 3. If the projection is not applied, p_d from (11) could beyond the support

FIGURE 3: Projection of p_d in support polygon.

polygon, and the CoM trajectory generated by p_d will cause falling on the ground.

As Figure 3 shows, we found a new projection ZMP p'_d in support polygon to make CoM follow the target direction of CP. The projection rules can be summarized as follows:

- (i) If p_d is located in support polygon, the target ZMP p_d is in the reachable region of the biped, then projection ZMP $p'_d = p_d$.
- (ii) If p_d is located outside the support polygon, p'_d is the closest point on the inner boundary of the support polygon to $(\xi_d - \xi)$.

In practical application, the number of reverse iterative footprint calculation F is limited. Based on the current foothold p_i and initial CP $\xi_{init,i}$, the position of ZMP can be solved on a limited horizon F . CoM mapping is used to make the CoM approach the target position. But as 3 shows, there is a discontinuity between current p and $p'_d = p_d$ which leads to discontinuous input to the walking pattern generator. To compensate for this problem, the MPC controller is introduced in the next section to track the CP trajectory on the one hand and export smooth CoM trajectory, so that the biped will not lose stability due to the sudden change of dynamic parameters in the process of walking.

2.3. Footprint Generator with Variable Velocity. The footprint generator module and MPC controller module are implemented synchronously. The input of the footprint generator is reference velocity v_x, v_y and angular velocity w around the z -axis. The output of this generator is the next

footprint position, direction, and time. With the change of input reference velocity, the cycle and step length of biped walking will be affected, and the footprint generator calculates the next foothold position by solving the quadratic optimization problem. The time and orientation of the foothold of the output of the footprint generator will be used as the reference input of the MPC controller in the next step.

As shown in Figure 1, the input of the footprint generator is v_x, v_y, ω from time t_k to $t_{k+P} = t_k + T_p$, where T_p is the duration of the footstep. The output of the footprint generator is the sequence $(\hat{X}_f^k, \hat{Y}_f^k, \Theta_f^k)$ on time sequence \mathcal{T}_s^k , and

$$\begin{aligned} \hat{\mathbf{X}}_f^k &= (x_f^1 \dots x_f^F)^T, \\ \hat{\mathbf{Y}}_f^k &= (y_f^1 \dots y_f^F)^T, \\ \Theta_f^k &= (\theta_f^1 \dots \theta_f^F)^T, \\ \mathcal{T}_s^k &= \{T_s^1, \dots, T_s^F\}, \end{aligned} \quad (13)$$

where F is the number of next footprints, and $(x_f^j, y_f^j, \theta_f^j)$ is the position and orientation of j -th footprint. T_s^j is the duration from $j-1$ -th to j -th footprint. Since the input velocity is variable, the prediction horizon of time T_s is constant, and $\sum_{j=1}^F T_s^j = T_p$, and then, distribution of $(x_f^j, y_f^j, \theta_f^j)$ is nonequidistant during T_p .

Assume that the biped is within the j -th support phase switching duration T_s^j , \bar{v} is average velocity, \bar{T}_s is step period duration, and \bar{L}_s is stride length, so $\bar{v} = \bar{L}_s / \bar{T}_s$. Walking velocity is limited by the biped dynamics such as CoM height, degree of freedom, and leg length. \bar{v} is determined by both T_s^j and \bar{L}_s . Δv is velocity small variation in Δt , and then,

$$v = \bar{v} + \Delta v = \frac{\bar{L}_s + \Delta L_s}{\bar{T}_s - \Delta T_s}, \quad (14)$$

where $\Delta L_s = \gamma \Delta T_s$, and (14) can be expressed as

$$T_s = \bar{T}_s \frac{\gamma + \bar{v}}{\gamma + v}. \quad (15)$$

Let $\bar{v} = 0.01m/s$, $\bar{T}_s = 0.8s$, and the relationship between velocity and duration can be shown in Figure 4.

During a small time interval δt , velocity can be approximately regarded as a linear variation. Therefore, the orientation angle of the robot can be approximately constant, and angular velocity is ignored in (14).

Considering time sequence $t_2^1, t_2^2, \dots, t_2^F$ iterative relation, the time iterative equation within $[t_k, t_k + T_p]$ is

$$t_s^j = t_s^{j-1} + \bar{T}_s \frac{\gamma + \bar{v}}{\gamma + v(t_s^{j-1})}, \quad (16)$$

where t_s^0 is the end time of the last support leg duration, and the iteration is over when $j > k + P$. Then, a time sequence $\mathcal{T}_s^k = \{T_s^1, \dots, T_s^F\}$ is outputted with $T_s^j = t_s^j - t_s^{j-1}$.

According to the omnidirectional motion model of a mobile robot, the bipedal moves to the target point at any angle and direction in the horizontal plane with v_x, v_y, ω .

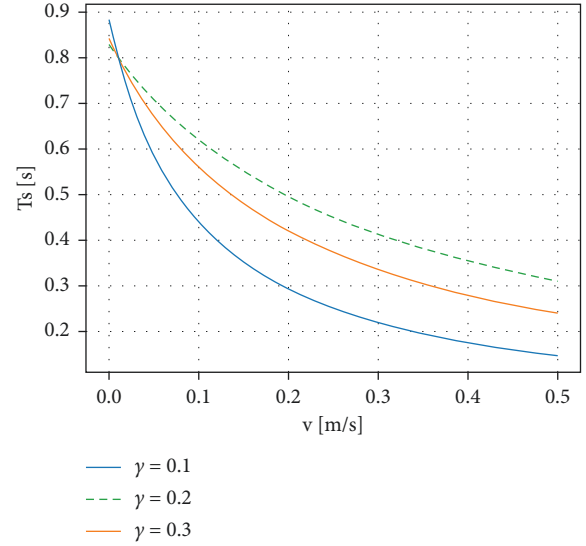


FIGURE 4: Rules for step duration T_s and velocity \bar{v} , and comparison of different γ .

$$\begin{pmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \end{pmatrix} = \begin{pmatrix} \cos\theta & -\sin\theta & 0 \\ \sin\theta & \cos\theta & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} v_x \\ v_y \\ \omega \end{pmatrix}. \quad (17)$$

The idea of the footprint generator is to allocate the future foothold under the condition of considering the constraints of kinematics and dynamics parameters of the biped in time $\mathcal{T}_s^k = \{T_s^1, \dots, T_s^F\}$. When the body root is subjected to push forces, according to (7), \dot{x} will change suddenly, which will affect the planned trajectory of x . The velocity and acceleration change suddenly, which affect the planning trajectory of CoM. The position of the foothold need be adjusted to make the CoM as close as possible near new CP so that the CoM can keep a gradually stable state.

The footprint planning can be transformed into two QP optimization problems as follows:

$$\begin{cases} \min_{\Theta_f^k} \sum_{j=1}^F \left(\theta_f^j - \theta_f^{j-1} - \int_{t_s^{j-1}}^{t_s^j} \omega(\tau) d\tau \right)^2 \text{ s.t. } |\theta_f^j - \theta_f^{j-1}| \leq \theta_{\max}, \end{cases} \quad (18)$$

where θ_{\max} is the maximum steering angle allowed for two consecutive footprints according to the physics dynamics of the biped. And the second QP problem is

$$\min_{\hat{x}_f^k, \hat{y}_f^k} \sum_{j=1}^F \left(\hat{x}_f^j - \hat{x}_f^{j-1} - \Delta x^j \right)^2 + \left(\hat{y}_f^j - \hat{y}_f^{j-1} - \Delta y^j \right)^2. \quad (19)$$

where the supporting foot position is $(\hat{x}_f^0, \hat{y}_f^0)$ at the starting time t_k . The output of the footprint generator $(\hat{X}_f^k, \hat{Y}_f^k, \Theta_f^k)$ is the input of the MPC controller, and the final ZMP and CoM trajectory is calculated by MPC. Figure 5 shows an example of footprint generation where the orientation of the robot coincides with the tangent of the motion path. The swing foot trajectory planning is simple, and in order to avoid sudden changes in the velocity of end point, we use Bezier curves to generate swing foot trajectory and then solve the joints rotation by inverse kinematics.

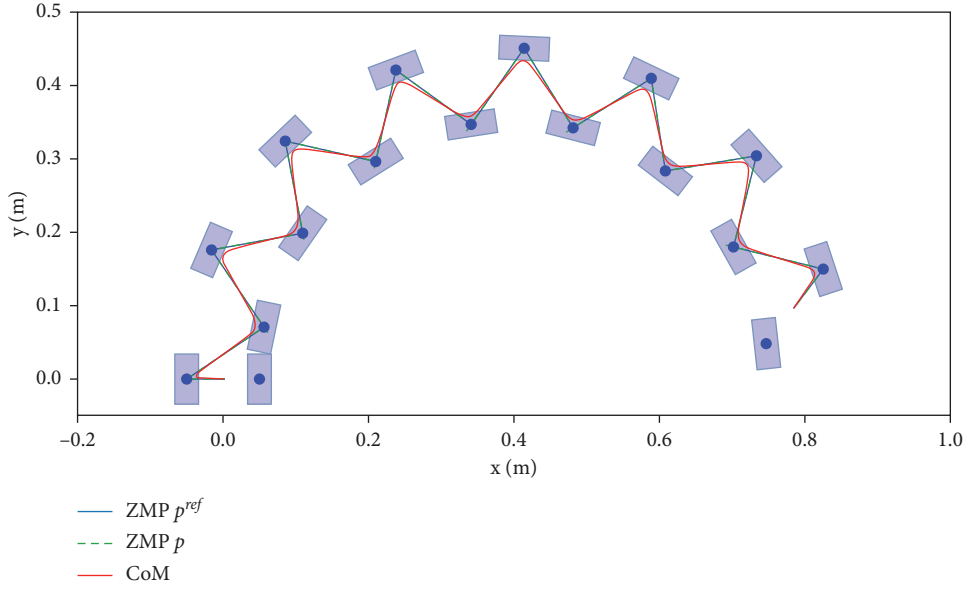


FIGURE 5: Candidate footsteps generated by the footprint generator in half circle demonstrate the omnidirectional feature.

2.4. MPC Controller. Predictive control approximates the robot to a three-dimensional linear inverted pendulum, introduces the state space method, optimizes the performance index by the linear quadratic regulator, and limits the movement of the CoM in the horizontal plane with constant height. The state feedback gain and predictive gain of the predictive controller are generated by solving discrete algebraic Riccati equations. Due to the consideration of state feedback, error feedback, and future target information, a stable walking mode can be generated with less computation.

The system only performs prediction of MPC controller at kT , then measures the actual state x, \dot{x} of the system as feedback by LIPM, and calculates the optimal control by

solving QP. N represents the prediction horizon of the MPC controller, and equation (6) can be rewritten within $[kT, (k + N)T]$.

$$\min_{u_k, \dots, u_{k+N}} \sum_{i=k}^{k+N-1} \frac{1}{2} \alpha (p_{i+1} - p_{i+1}^{ref})^2 + \frac{1}{2} \beta u_i^2. \quad (20)$$

The above equation (20) is solved by solving complex algebraic Riccati equation in [5], but this method has low computational efficiency. Within MPC prediction horizon, equation (4) can be transformed to matrix operation. The state transition equation is rewritten with length N as

$$\begin{aligned} \begin{bmatrix} p_{k+1} \\ \vdots \\ p_{k+N} \end{bmatrix} &= \begin{bmatrix} 1 & T & T^2/2 - 1/w_0^2 \\ \vdots & \vdots & \vdots \\ 1 & NT & N^2T^2/2 - 1/w_0^2 \end{bmatrix} \hat{x}_k \\ &+ \begin{bmatrix} T^3/6 - T/w_0^2 & 0 & 0 \\ \vdots & \ddots & 0 \\ (1 + 3N + 3N^2)T^3/6 - T/w_0^2 & \dots & T^3/6 - T/w_0^2 \end{bmatrix} \\ &\times \begin{bmatrix} u_k \\ \vdots \\ u_{k+N-1} \end{bmatrix}, \end{aligned} \quad (21)$$

And (21) can be abbreviated as

$$P_{k+1} = Q_x \hat{x}_k + R_u U_k. \quad (22)$$

According to (20), the QP equation can be represented as

$$\min_{x_k} \frac{1}{2} \alpha (P_{k+1} - P_{k+1}^{ref})^2 + \frac{1}{2} \beta U_k^2, \quad (23)$$

And a standard QP form can be generated by substituting (22) into (23)

$$\min_{U_k} \left[\frac{1}{2} (\alpha R_u^2 + \beta) U_k^2 + \alpha (Q_x \hat{x}_k R_u - P_{k+1}^{ref} R_u) U_k \right]. \quad (24)$$

According to the definition of MPC, u_k at the current moment is the first prediction of MPC prediction trajectory; that is,

$$u_k = e^T U_k, \quad (25)$$

where $e^T = [1, 0, \dots, 0]^T$, and according to the state formula of LIPM (4), the trajectory of x and output position of ZMP at the next moment are calculated as

$$\hat{x}_{k+1} = A\hat{x}_k + Bu_k = A\hat{x}_k + Be^T U_k. \quad (26)$$

Within the predictive horizon NT , a balance between response speed and control cost of system can be found by adjusting α/β ($\alpha > 0, \beta > 0$), and in this article, $\alpha/\beta = 10^{-5}$. When the system and environment models are well known, NT can approach infinity. But LIPM model itself is a linear approximation to the walking dynamics of biped robot with systematic model errors, and environment cannot be accurately modeled. So prediction horizon could not be infinite. In fact, when N increases, the gain of the system is close to 0, which means that the influence of the predicted future value on the stability of the current system is reduced, but it will significantly decrease the computational efficiency of QP calculation. Therefore, this paper sets the MPC control horizon to $NT = 1s$.

3. Results

The performance of the proposed dynamic balance control framework in Figure 1 is proved by simulation experiment, we present simulation results using the LIPM and MPC simulation in bullet environment, the input velocity is constant, the single leg support phase time is constant $t_{sup} = 0.5s$, supporting leg switching time $t_{sw} = 0.33s$, MPC horizon $N = 30$, and the minimum control frequency is 100 Hz. The external push force $f_{ext} = 0$, and the robot is not subject to external interference, so that walks along a straight line with $N_{step} = 6$, and according to the foothold generator, the robot's foothold position is shown in Figure 6.

Since the LIPM is decoupled from each other in X and Y directions, body CoM and ZMP are solved separately by the MPC controller, as is shown in Figure 7. Although there is slight noise jitter in the output ZMP trajectory, the CoM maintains smooth transition, and there is no big speed mutation when the support legs are exchanged. Therefore, the MPC method has good tracking performance.

For the omnidirectional walking, Figure 5 shows velocity direction of walking can be changed in real time by the footprint generator, and $\theta_{max} = \pi/4$. The robot walks around the center of the circle along with a radius of 2m. The system receives high-level control input v_x, v_y, w , and the optimization problems (18) and (19) is solved at each step. By

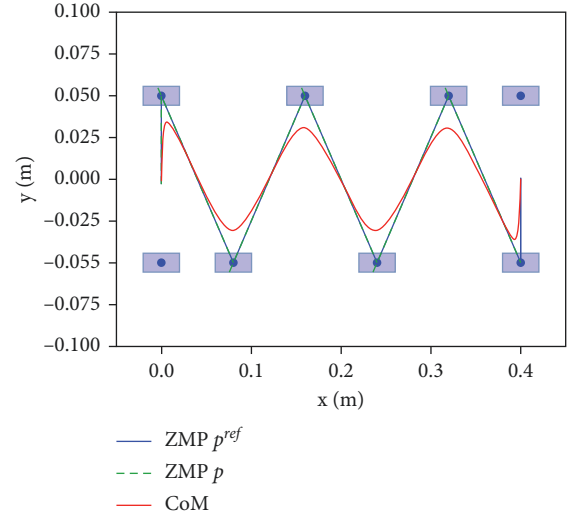


FIGURE 6: The CoM and ZMP trajectory when walking in a straight line $N_{step} = 6$.

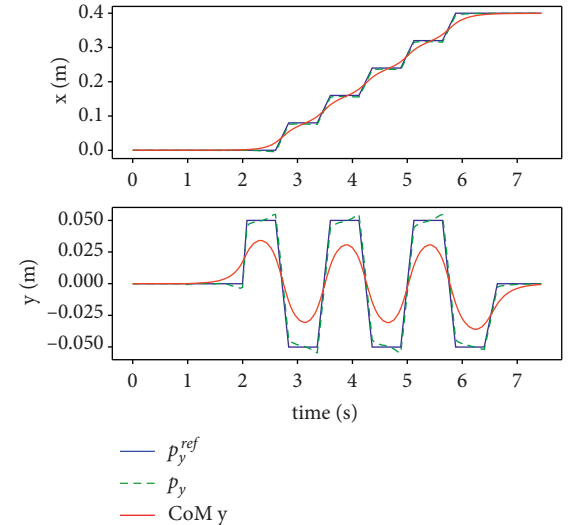


FIGURE 7: When walking in a straight line, the CoM in the X direction (up) and Y direction (down) and ZMP trajectory.

solving the inverse kinematics, the next step foothold rotation is realized by the hip joint rotation; thus, the omnidirectional walking on the horizon plane of the biped is implemented.

The performance of the proposed method is measured in case of external disturbance in x-y plane, as shown in Figures 8–11. An instantaneous external thrust $f_{ext} = 10N$ is added to the CoM of the robot between 4s and 5s, that duration is 0.01 s. It can be seen that the system can adjust the foothold position in 1 step in both directions and be stabilized.

Since the interference of external thrust, the biped will deviate from the predetermined trajectory, but the velocity and direction of biped walking can be changed in real time, and the system has the ability to adjust the input parameters to make the biped walking towards the target position.

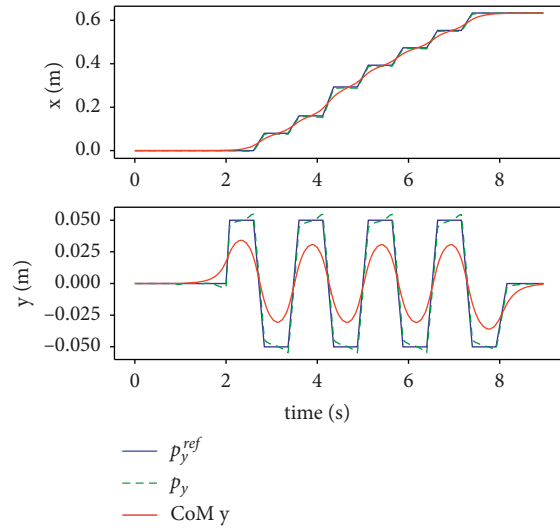


FIGURE 8: Simulation of tracking the trajectory of time with disturbance force in the sagittal direction.

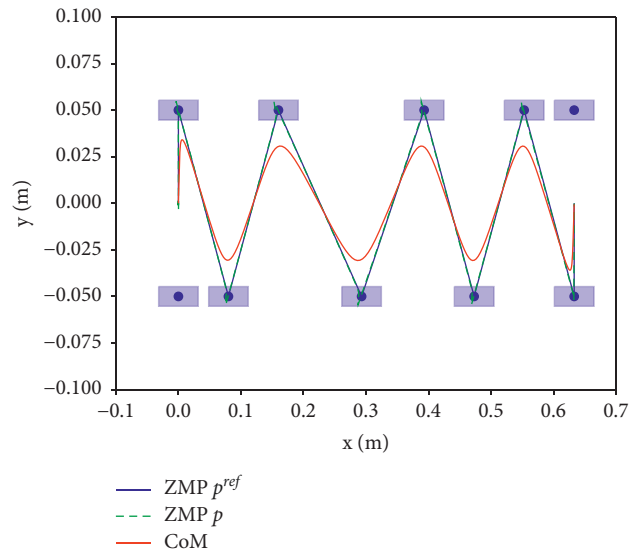


FIGURE 9: Simulation of tracking the trajectory in horizon plane with disturbance force in the sagittal direction.

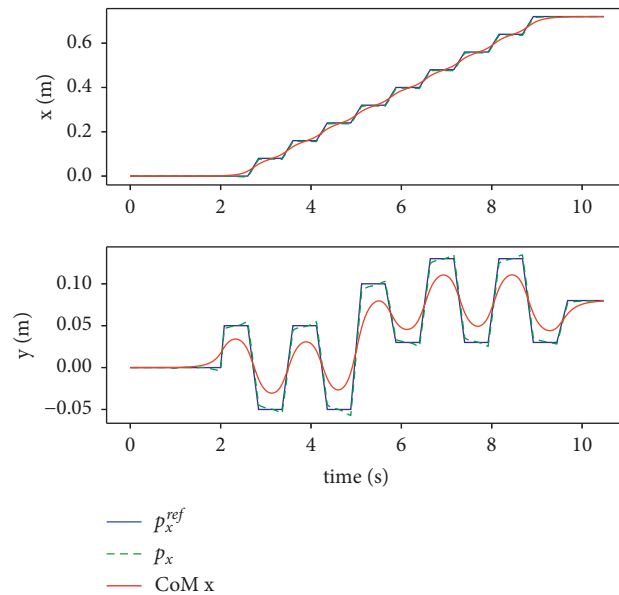


FIGURE 10: Simulation of tracking the trajectory of time with disturbance force in the coronal direction.

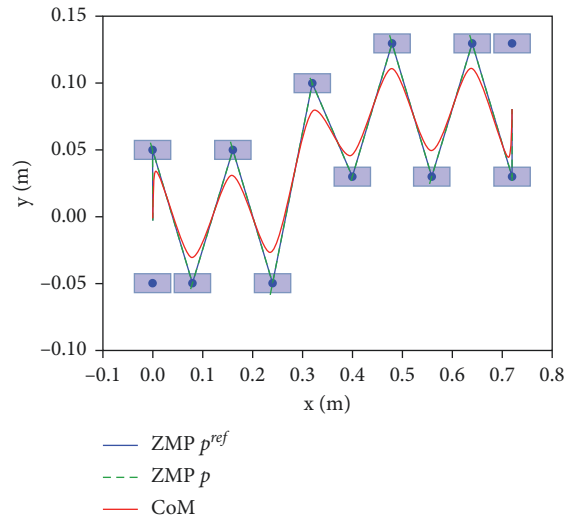


FIGURE 11: Simulation of tracking the trajectory in horizon plane with disturbance force in the coronal direction.

4. Conclusions

At the first, a simplified LIPM is used to linearize the dynamic model of biped walking, MPC is used as a gait pattern generator to keep dynamics and kinematics feasibility, and CoM and ZMP trajectory of the biped is solved and predicted in a limited horizon. Since the bipedal system is an unstable system, it is easy to be disturbed by external thrust, resulting in system instability and falling down. Therefore, this paper designs a bipedal balance control method considering capture point with MPC controller, which realizes the footprints planning with variable speed, so that the robot can walk in omnidirectional directions on the plane. At the same time, the capture point parameters of the system are introduced as feedback. When the system is disturbed by external thrust, the acquisition point will change suddenly. By controlling the ZMP point trajectory, the centroid can track the direction of the capture point, and the system will walk to keep it stable. However, CP may fall outside the support polygon. We use a projection method to make the next landing footprint always fall in the range of the biped reachable area and reach the target CP in an incremental manner. For solving the problem of ZMP input point discontinuity caused by projection, the system uses the MPC method to predict the smooth trajectory curve of centroid in the future according to the new foothold position. The position of joint of the full body is calculated by inverse kinematics and drives the biped realized stable walking.

We only focus on the dynamic walking of bipedal on the horizontal plane. It is necessary to analyze the dynamic walking of bipedal on the nonstationary ground and extend the planning of foothold in a two-dimensional plane to three-dimensional space. On the other hand, the gait pattern generator is mainly a linear inverted pendulum model, which is an approximation of bipedal walking and can not plan complex action behavior. And there will be some improvements in the system approximation model, and a more accurate dynamic and environmental model will be used to predict and analyze the robot's future actions and behaviors.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

The authors thank Leju Robotics for providing funding and resources for this project. This research was supported in part by the basic research (T01) project of the Shenzhen Science and Innovation Commission (JCYJ20180307124010740).

References

- [1] C.-F. Juang and Y.-T. Yeh, "Multiobjective evolution of biped robot gaits using advanced continuous ant-colony optimized recurrent neural networks," *IEEE Transactions on Cybernetics*, vol. 48, no. 6, pp. 1910–1922, 2017.
- [2] P.-B. Wieber, R. Tedrake, and S. Kuindersma, "Modeling and control of legged robots," *Springer Handbook of Robotics*, pp. 1203–1234, 2016.
- [3] X. Yang, H. She, H. Lu, T. Fukuda, and Y. Shen, "State of the art: bipedal robots for lower limb rehabilitation," *Applied Sciences*, vol. 7, no. 11, p. 1182, 2017.
- [4] S. Kajita, K. Fumio, K. Kenji, Y. Kazuhito, and H. Hirohisa, "The 3D linear inverted pendulum mode: a simple modeling for a biped walking pattern generation," vol. 1, pp. 239–246, in *Proceedings of the 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 1, pp. 239–246, IEEE, Maui, HI, USA, 29 Oct.-3 Nov. 2001.
- [5] J. Park and Y. Youngil, "General ZMP preview control for bipedal walking," in *Proceedings of the 2007 IEEE International Conference on Robotics and Automation*, pp. 2682–2687, IEEE, Rome, Italy, 10-14 April 2007.

- [6] J.-w. Luo, Y.-I. Fu, and S.-g. Wang, "3D stable biped walking control and implementation on real robot," *Advanced Robotics*, vol. 31, no. 12, pp. 634–649, 2017.
- [7] Z. Yu, X. Chen, Q. Huang et al., "Gait planning of omnidirectional walk on inclined ground for biped robots," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 46, no. 7, pp. 888–897, 2016.
- [8] L. Lanari, U. Oliver, H. Seth, and S. Ingmar, "Boundedness approach to gait planning for the flexible linear inverted pendulum model," in *Robot World Cup*, pp. 58–70, 2016.
- [9] X. Li, L. Yangmin, and C. Xinzhe, "Kinematic analysis and gait planning for a Darwin-OP Humanoid Robot," in *Proceedings of the In 2016 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, pp. 1442–1447, IEEE, Qingdao, China, 3-7 Dec. 2016.
- [10] H.-M. Joe and J.-H. Oh, "Balance recovery through model predictive control based on capture point dynamics for biped walking robot," *Robotics and Autonomous Systems*, vol. 105, pp. 1–10, 2018.
- [11] H. Gritli and S. Belghith, "Identification, stability and stabilization of limit cycles in a compass-gait biped model via a hybrid poincaré map," *Advances and Applications in Non-linear Control Systems*, pp. 259–289, 2016.
- [12] J. Pratt, C. John, D. Sergey, and G. Ambarish, "Capture point: a step toward humanoid push recovery," in *Proceedings of the In 2006 6th IEEE-RAS international conference on humanoid robots*, pp. 200–207, IEEE, Genova, Italy, 4-6 Dec. 2006.
- [13] J. Engelsberger, O. Christian, A. R. Máximo, A.-S. Alin, and H. Gerhard, "Bipedal walking control based on capture point dynamics," in *Proceedings of the 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 4420–4427, IEEE, San Francisco, CA, USA, 25-30 Sept. 2011.
- [14] D. Tran, H. Fred, and N. John, "A humanoid robot learns to recover perturbation during swinging motion," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 10, pp. 3701–3712, 2018.
- [15] M. Missura and B. Sven, "Gradient-driven online learning of bipedal push recovery," in *Proceedings of the In 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 387–392, IEEE, Hamburg, Germany, 28 Sept.-2 Oct. 2015.
- [16] M. Parsa and F. Mohammad, "Robust nonlinear model predictive trajectory free control of biped robots based on nonlinear disturbance observer," in *Proceedings of the 2010 18th Iranian Conference on Electrical Engineering*, pp. 617–622, IEEE, Isfahan, Iran, 11-13 May 2010.
- [17] F. M. Smaldone, S. Nicola, M. Valerio, L. Leonardo, and O. Giuseppe, "Gait generation using intrinsically stable MPC in the presence of persistent disturbances," in *Proceedings of the 2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids)*, pp. 651–656, IEEE, Toronto, ON, Canada, 15-17 Oct. 2019.
- [18] H. Diedam, D. Dimitar, W. Pierre-Brice, M. Katja, and D. Moritz, "Online walking gait generation with adaptive foot positioning through linear model predictive control," in *Proceedings of the 2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 1121–1126, IEEE, Nice, France, 22-26 Sept. 2008.
- [19] A. Herdt, P. Nicolas, and W. Pierre-Brice, "Walking without thinking about it," in *Proceedings of the 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 190–195, IEEE, Taipei, Taiwan, 18-22 Oct. 2010.
- [20] A. Herdt, D. Holger, W. Pierre-Brice, D. Dimitar, M. Katja, and D. Moritz, "Online walking motion generation with automatic footstep placement," *Advanced Robotics*, vol. 24, no. 5–6, pp. 719–737, 2010.
- [21] B. Amos, D. J. R. Ivan, S. Jacob, B. Byron, and J. Zico Kolter, "Differentiable mpc for end-to-end planning and control," arXiv preprint: <http://arXiv.org/abs/1810.13400>, 2018.
- [22] N. Scianca, D. De. Simone, L. Lanari, and G. Oriolo, "MPC for humanoid gait generation: stability and feasibility," *IEEE Transactions on Robotics*, vol. 36, no. 4, pp. 1171–1188, 2020.
- [23] C.-F. Juang and Y.-T. Yeh, "Multiobjective evolution of biped robot gaits using advanced continuous ant-colony optimized recurrent neural networks," *IEEE Transactions on Cybernetics*, vol. 48, no. 6, pp. 1910–1922, 2018.