Research Article

Multiphase Trajectory Generation for Planar Biped Robot Using Direct Collocation Method

Yubin Liu ⓘ, **Shuai Heng** ⓘ, **Xizhe Zang** ⓘ, **Zhenkun Lin** ⓘ, **and Jie Zhao** ⓘ

State Key Laboratory of Robotics and System, Harbin Institute of Technology, Harbin 150080, China

Correspondence should be addressed to Xizhe Zang; zangxizhe@hit.edu.cn

Stability and energy efficiency are the main focuses in the bipedal robot field. In this paper, we apply a multiphase gait, which is different from the widely used two-phase gait, to improve the stability at the moment, when a biped robot transfers from the double support phase to the single support phase. Then, we create dynamic equations with contact forces in each phase using Lagrangian formulation. Furthermore, the direct collocation method is utilized to generate the optimal trajectory toward both stability and energy efficiency. Finally, the comparison between multiphase gait and two-phase gait is performed with numerical simulations. The results prove that multiphase gait increases the stability margin in the cost of slightly decreasing energy efficiency. Besides, both gaits show a similar human-like characteristic in hip height variation during walking.

1. Introduction

In the field of bipedal robot research, both stability and energy efficiency are always the main goals. At present, the bipedal robot still experiences low stability and high energy cost problem. This paper aims to dig toward this orientation further.

Raibert developed a successful running bipedal robot with one-foot gait, which distinguishes the state of the leg into two phases, support or stance phase and flight phase [1]. The support phase is when the leg provides support, and the flight phase is when the leg is in the air. The leg switches two phases in strict alternation. As for the walk of the bipedal robot, most researchers suppose it consists of two successive phases, single support phase (SSP) and double support phase (DSP). These two phases are easily understood according to the above definition of support phase: SSP is a period when only one leg is in the support phase, and DSP is when both legs are in the support phase. Actually, the studies of human walking gait find that there are two subphases of DSP and SSP each [2]. Then, these two foot rotation subphases are applied to biped robot [3–5]. The degree of freedom (DoF) of the biped robot, which is used in this paper, varies in different phases. In SSP, the robot is fully actuated, and when

considering the heel lift subphase, it is underactuated; in DSP, it is overactuated [6]. In this paper, we add the foot rotation subphase in DSP based on the two-phase gait, so there are only fully actuated SSP and overactuated DSP. The theoretical benefit of this gait is illustrated in Section 2.2. The finding of the pros and cons of this multiphase gait is our focus in this paper. This gait is close to that used in paper [7], except we use different methods to generate trajectory and give out the comparison with two-phase gait to analyze the benefit of this multiphase gait.

Energy efficiency and stability have long been objectives of trajectory generation. For stability, the zero moment point (ZMP) [8] has been used since the last century. The ZMP is the point on the ground, where the aggregation of moments about this point is zero. If ZMP is within the convex sets consisting of regions where feet contact the ground, the robot feet would not rotate around ZMP, and the robot is considered stable. This stability criterion has been widely applied in biped like ASIMO, NAO, and HRP-2. There is a different method called hybrid zero dynamics (HZD) [9], which can be used to analyze the stability of periodic orbits. This method is good at solving underactuated systems, but comparing to the ZMP method, this method is relatively complex and hard to understand. Besides, we do not

consider the underactuated phase here; thus, we choose the ZMP method in this paper.

For optimal trajectory generation, the most frequently used method is parameter optimization, which firstly represents the joint trajectories as functions of time decided by a vector of real numbers and then takes these numbers as the decision variables of the optimization problem. Different gradient-free optimization methods, which do not need gradients to determine the search direction, are utilized to find these decision variables. Farzdpour et al. [7] used the genetic algorithm (GA) to obtain the key parameters in trajectory generation for a seven-link planar bipedal robot with multiphase. Elhosseini et al. [10] used A-C parametric whale optimization algorithm (WOA) to find optimal parameters of the hip trajectory making ZMP close to the middle of the support polygon in two-phase gait. This method is an efficient and tractable way of generating suboptimal trajectories. However, since this method reduced discrete optimization variables to a finite number, path constraints, which are along the whole trajectory period, like dynamic equation constraints, are difficult to satisfy. Besides, polynomial functions may arise undesirable oscillations of optimized functions or jerky variations at connecting points [11].

Some researchers applied the differential dynamic programming (DDP) method on trajectory generation. Feng et al. [12] achieved online trajectory optimization by using the DDP method to derive the trajectory of the center of mass of the robot as a high-level trajectory. Budhiraja et al. [13] used DDP with Karush–Kuhn–Tucker constraint to generate the whole-body trajectory while tracking the centroidal trajectory. This method usually combines with a simplified model and can generate the trajectory in milliseconds so it can be applied online. However, the DDP approach needs the second-order derivatives of the system dynamics, which leads to the issue that is unsuitable for large problems, and does not have a unified solution for the problem with general inequality constraints [14].

Except for the above parametric optimization and DDP method, the direct collocation method [15–17] is one promising alternative to effectively solve trajectory optimization problems [18]. It is especially suitable for highly complex problems. Besides, it can be solved with present nonlinear programming (NLP) solvers, which can converge with poor initial guesses and are extremely computationally efficient [19]. Most importantly, it overcomes some limitations of the above methods since it can satisfy a bunch of path constraints and do not need the second-order derivatives of the system dynamics, which is hard to derive for the complex full-dimensional dynamic equations.

There are many works about the direct collocation method applied in the legged robot area. In 1999, Hardt et al. [20] implemented direct collocation method using software DIRCOL to generate minimum energy symmetric, periodic gaits for a five-link bipedal robot with SSP, and instantaneous DSP gait. Xi et al. [21] used both direct collocation and multiple shooting methods to figure out how increasing speed affects the choice of gait and also inverse, under the objective of minimizing the cost of transport for both a

planar biped and planar quadruped. Multiple shooting method is one of the trajectory optimization methods. Compared to the direct collocation method, this method has benefits like simplicity and straightforward, but it suffers slow convergence speed and is sensitive to initial guesses [22]. Ma et al. [23] used nonlinear programming toolbox FROST [24] to generate walking gait on the slippery surface combining with the hybrid zero dynamics control method to achieve stable walking for the AMBER-3M planar robot with point feet. Li et al. [25] utilized a direct collocation method with the closed-form of centroidal momentum (CM), which is CM generated by ground reaction forces (GRFs) that should match with CM calculated from the robot's generalized coordinates and velocities, to generate the trajectory for a five-link legged robot with the reduced numerical error.

This paper aims to testify the benefits of three phases (SSP, DSP subphase1, and DSP subphase2) walking style and to generate the optimal trajectory for our seven-link planar biped robot toward energy efficiency and stability for this walking style. In this paper, we use the direct collocation method to generate the optimal trajectory for this walking style, which can increase the ZMP stability margin when biped robot transfers from DSP to SSP. The holonomic dynamic equations with contact forces in each phase are created by the Lagrangian method, while the contact impulse is not considered in our paper to keep the velocity continuity. Various constraints are considered in our optimization, like unilateral contact, friction cone, and given speed. We compare the generated two trajectories (three-phase gait and two-phase gait) in energy efficiency and stability and find that three-phase gait increases the stability although it is in the cost of slight energy cost, and direct collocation method brings about the human-like hip height trajectory during SSP, which is the cycloid curve.

The rest of the paper is organized as follows. Section 2 illustrates the differences between the two-phase walking style and three-phase walking style. The procedure for deriving the multiphase dynamic model is also presented. Section 3 is about the implementation details of direct collocation for generating optimal trajectory. The objectives and constraints of optimization are provided. Section 4 presents the generated three-phase and two-phase gaits. The comparison between these two gaits is conducted. In the end, conclusions and discussion are presented in Section 5.

2. Multiphase Biped Model

2.1. Model Symbol Definition. The bipedal robot in this study is a seven-link planar biped robot, which owns six actuators. Every two adjacent links are connected by an actuated revolute joint. Figure 1 presents the schematic biped robot. The symbols' meanings and values are given in Table 1, and they are consistent in the paper. Most of the parameters match with our biped robot [26] except the parameters of torso since our robot now does not have the upper body. We add it here because the upper body can equilibrate the angular momentum of the swing leg. The parameters of the upper body, mainly the mass and the length, are calculated
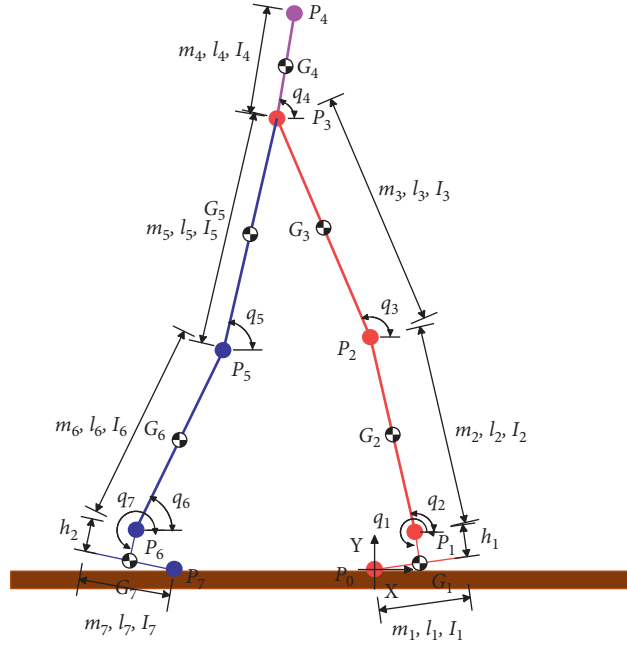
Figure 1: Biped robot model.

Table 1: Physical parameters of the biped model.

Symbol	Value	Name
m_1, m_7	0.75 kg	Mass of foot
m_2, m_6	1 kg	Mass of tibia
m_3, m_5	2.5 kg	Mass of femur
m_4	21.25 kg	Mass of torso
l_1, l_7	0.16 m	Length of foot
h_1, h_2	0.035 m	Height of foot
r_1, r_2	0.5	Ratio of forefoot
l_2, l_6	0.298 m	Length of tibia
l_3, l_5	0.279 m	Length of femur
l_4	0.5678 m	Length of torso
I_1, I_7	0.0025 kgm^2	Rotational inertia of foot
I_2, I_6	0.0296 kgm^2	Rotational inertia of tibia
I_3, I_5	0.0649 kgm^2	Rotational inertia of femur
I_4	0.5709 kgm^2	Rotational inertia of torso

according to the ratio between the upper body and lower body in paper [27].

Forefoot is the length from toe to ankle. Inertia is about its center of mass.

There are other data in Figure 1 needed to be illustrated. The fixed coordinate is set in the heel of the support leg (in red). The direction of the X-axis is along the walk direction, and the direction of the Y-axis is vertical to the ground and point to the top. $\mathbf{q} = [q_1, q_2, q_3, q_4, q_5, q_6, q_7]$ are the generalized coordinates used for depicting the configuration of robot. They are measured from the horizontal line through the revolute point to the link, which is called absolute angle usually, and the positive direction is counterclockwise same with convention. P_0, P_1, P_2, etc. are the position of revolute joints, which can be derived by $\mathbf{q}$ and robot parameters through kinematics. G_1, G_2, G_3, etc. are points of center of mass (CoM) of each link. Usually, we set CoM in the

geometry middle point of the link except for the foot we put it in the middle of the sole.

2.2. The Biped Robot Dynamic Model. Before we start to derive the dynamic model, we need to illustrate "multiphase" in the heading. In this paper, we only consider periodic gait. A gait is defined from one foot off the ground until the other foot leaves the ground.

In this paper, we consider two different walking styles shown in Figure 2. The upper one includes two phases: SSP (right half) and DSP (left half), and this walking style is widely used. To make it more clear, we picture the middle state in SSP and DSP. The single support phase is when one leg (swing leg) of the robot swings, and the other leg (support leg) is on the ground. The double support phase is the state when both legs are touching the ground. When using ZMP as a stability criterion, we need ZMP to stay within the support polygon (SP), which is the convex sets consisting of regions where feet contact with the ground, and the minimum distance from ZMP to the boundary of SP is called the stability margin. The more the stability margin, the more stable the robot. Since we limit our research to planar walk, the support polygon is a line along the X-axis and ZMP is a point in the X-axis.

For the upper walking style, in DSP, the SP is from x_A to x_B, where x_A is the horizontal position of swing leg toe and x_B is the position of support leg heel. In SSP, the SP is from x_B to x_C, where x_C is the horizontal position of support leg toe. So, when the robot switches from DSP to SSP, ZMP must go through the intersection point x_B ($x_B = [x_A \ x_B] \cap [x_B \ x_C]$); i.e., the SP is only a point; thus, at this moment, the stability margin is zero, a small deviation will cause the robot to fall. To prevent this situation, the below walking style is given in Figure 2. This walking style divided the double
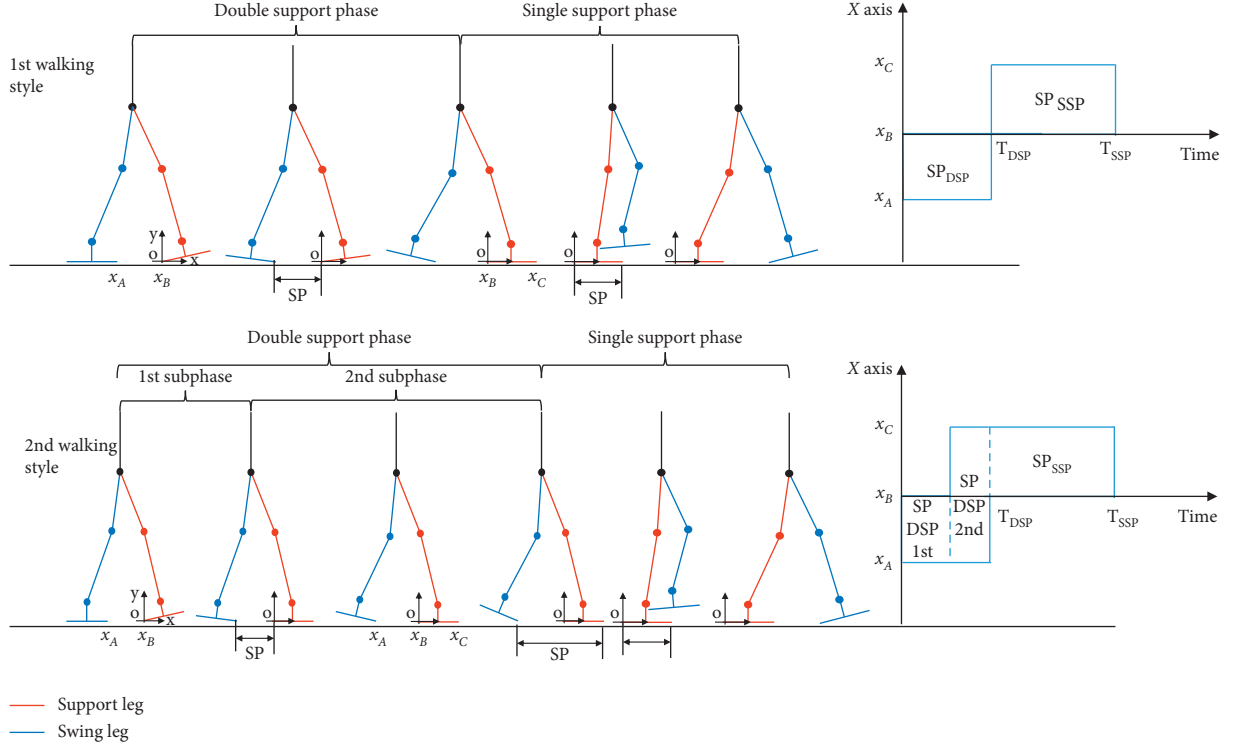
FIGURE 2: Two kinds of walking styles.

support phase into two subphases (subphase1 and sub-phase2). In subphase1, the support foot rotates about its heel to flat, while the swing foot rotates about its toe to an intermediary angle, and its SP is from x_A to x_B. In subphase2, the support foot keeps flat and the swing foot continues rotating until toe-off, which is the initial state of SSP and its SP is from x_A to x_C. Therefore, at the moment, when robot switches from DSP subphase2 to SSP, the stability margin is a line between x_B and x_C ($[x_B \, x_C] = [x_A \, x_C] \cap [x_B \, x_C]$). Comparing to the first walking style, the stability margin at transferring moment increases.

Note when the robot transfers from SSP to DSP, the swing leg (in blue) becomes the support leg (in red), likewise in original support leg.

The following is the procedure of deriving the equation of motion (EoM) for each phase. To simplify our procedure, we make the following modeling assumptions:

The support leg foot is in full contact with the ground during SSP, and there is no slippage.

During DSP, the support leg foot and swing foot rotate around a pin point, which does not slip.

The instantaneous impact event, when swing foot contacts with ground, is avoided by making the contact velocity zero.

2.2.1. Single Support Phase. Multibody dynamics with contact usually formulate as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{b}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} + \mathbf{J}^T\boldsymbol{\lambda}, \qquad (1)$$

where $\mathbf{M}(\mathbf{q}) \in \mathbf{R}^{n_q \times n_q}$ is the generalized mass matrix; n_q is the dimension of $\mathbf{q}$; $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}} \in \mathbf{R}^{n_q}$ are the generalized position, velocity, and acceleration, respectively; $\mathbf{b}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbf{R}^{n_q \times n_q}$ is the Coriolis and centrifugal matrix; $\mathbf{g}(\mathbf{q}) \in \mathbf{R}^{n_q}$ is the gravitational term; $\boldsymbol{\tau} \in \mathbf{R}^{n_q}$ is the generalized actuator forces; $\mathbf{J} \in \mathbf{R}^{n_c \times n_q}$ is the Jacobian matrix of contact points; n_c is the number of contact points; and $\boldsymbol{\lambda} \in \mathbf{R}^{n_c}$ is the Cartesian contact forces.

Then, we use the Euler–Lagrange formalism to derive the EoM of the single support phase. Since, in SSP, the link 1 does not move, i.e., can be regarded as the ground, we can eliminate link 1 and its corresponding q_1. In the end, the EoM of single support phase can be written as

$$\mathbf{M}_1(\mathbf{q})_{(6\times6)}\ddot{\mathbf{q}} + \mathbf{b}_1(\mathbf{q}, \dot{\mathbf{q}})_{(6\times6)}\dot{\mathbf{q}} + \mathbf{g}_1(\mathbf{q})_{(6\times1)} = \mathbf{B}_{1(6\times6)}\mathbf{u}, \quad (2)$$

where $\mathbf{u} = [u1; u2; u3; u4; u5; u6]$ is the vector of actuator torques and $\mathbf{B}_1$ is the distribution matrix, which maps the actuator torques to generalized actuator forces. The actual dimension of matrices is illustrated in the bracket.

2.2.2. Double Support Subphase1. In DSP, the biped robot is overactuated because there are two contacts (ground with support leg and swing leg). In kinematics, the robot is the closed chain, and contacts can be handled as kinematics constraints. In subphase1, the heel of support leg $\mathbf{P}_0$ and toe of swing leg $\mathbf{P}_7$ are both in contact with the ground. Suppose the distance between $\mathbf{P}_0$ and $\mathbf{P}_7$ is D_s. We use P_{0x}, P_{7x} and P_{0y}, P_{7y} represent the X coordinate and Y coordinate of each point. Then, the holonomic kinematic constraints are as follows:

$$\mathbf{\Omega}_1(\mathbf{q}) = \begin{bmatrix} P_{0x} - P_{7x} - D_s \\ P_{0y} - P_{7y} \end{bmatrix} = 0. \tag{3}$$

The Jacobian of holonomic constraints is

$$\mathbf{J}_{c1}(\mathbf{q}) = \frac{\partial \mathbf{\Omega}_1(\mathbf{q})}{\partial \mathbf{q}}. \tag{4}$$

Hence, the EoM of the robot in subphase1 formulates as follows:

$$\mathbf{M}_2(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{b}_2(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}_2(\mathbf{q}) = \mathbf{Bu} + \mathbf{J}_{c1}(\mathbf{q})\boldsymbol{\lambda}_1. \tag{5}$$

$\boldsymbol{\lambda}_1$ is unknown here, so we could not solve this equation now. One way to solve it is eliminating the $\boldsymbol{\lambda}_1$. First, we assume $\mathbf{J}_{c1}$ is the full rank, so there is the orthogonal complement matrix of it, $\mathbf{J}_{c1null}$, satisfying $\mathbf{J}_{c1}\mathbf{J}_{c1null} = \mathbf{J}_{c1null}^T \mathbf{J}_{c1}^T = 0$. Now we transfer to derive $\mathbf{J}_{c1null}$. We divide the generalized coordinates into two categories: independent ($\mathbf{q}_{in}$) and dependent coordinates ($\mathbf{q}_{de}$). In this phase, we can choose them as

$$\begin{cases} \mathbf{q}_{in}, & [q_1 \ q_2 \ q_3 \ q_4 \ a_5]^T, \\ \mathbf{q}_{de}, & [q_6 \ q_7]^T. \end{cases} \tag{6}$$

Taking the derivative of equation (3), we gain

$$\mathbf{J}_{c1}(\mathbf{q})\dot{\mathbf{q}} = 0. \tag{7}$$

Then, we combine equations (6) and (7):

$$\begin{bmatrix} \mathbf{J}_{c1in} & \mathbf{J}_{c1de} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{q}}_{in} \\ \dot{\mathbf{q}}_{de} \end{bmatrix} = \mathbf{J}_{c1in}\dot{\mathbf{q}}_{in} + \mathbf{J}_{c1de}\dot{\mathbf{q}}_{de} = 0, \tag{8}$$

where $\mathbf{J}_{c1in} = (\partial\Omega_1/\partial\mathbf{q}_{in})$ and $\mathbf{J}_{c1de} = (\partial\Omega_1/\partial\mathbf{q}_{de})$.

Then, we shift $\dot{\mathbf{q}}_{de}$ to the left side yield:

$$\dot{\mathbf{q}}_{de} = -\mathbf{J}_{c1de}^{-1}\mathbf{J}_{c1in}\dot{\mathbf{q}}_{in}. \tag{9}$$

Thus, $\dot{\mathbf{q}}$ can be expressed as

$$\dot{\mathbf{q}} = \begin{bmatrix} \dot{\mathbf{q}}_{in} \\ \dot{\mathbf{q}}_{de} \end{bmatrix} = \begin{bmatrix} \mathbf{I} \\ -\mathbf{J}_{c1de}^{-1}\mathbf{J}_{c1in} \end{bmatrix} \dot{\mathbf{q}}_{in}. \tag{10}$$

Replacing $\dot{\mathbf{q}}$ in equation (7),

$$\mathbf{J}_{c1}(\mathbf{q}) \begin{bmatrix} \mathbf{I} \\ -\mathbf{J}_{c1de}^{-1}\mathbf{J}_{c1in} \end{bmatrix} \dot{\mathbf{q}}_{in} = 0. \tag{11}$$

Given that $\dot{\mathbf{q}}_{in} \neq 0$, we can derive

$$\mathbf{J}_{c1}(\mathbf{q}) \begin{bmatrix} \mathbf{I} \\ -\mathbf{J}_{c1de}^{-1}\mathbf{J}_{c1in} \end{bmatrix} = 0,$$

$$\mathbf{J}_{c1null} = \begin{bmatrix} \mathbf{I} \\ -\mathbf{J}_{c1de}^{-1}\mathbf{J}_{c1in} \end{bmatrix}. \tag{12}$$

Now $\mathbf{J}_{c1null}$ is derived. Thus, we can multiply both sides of equation (5) with it:

$$\mathbf{J}_{c1null}^T\mathbf{M}_2(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{J}_{c1null}^T\mathbf{b}_2(\mathbf{q}, \dot{q})\dot{q} + \mathbf{J}_{c1null}^T\mathbf{g}_2(\mathbf{q})$$
$$= \mathbf{J}_{c1null}^T\mathbf{Bu} + \mathbf{J}_{c1null}^T\mathbf{J}_{c1}(\mathbf{q})\boldsymbol{\lambda}_1 \tag{13}$$
$$= \mathbf{J}_{c1null}^T\mathbf{Bu}.$$

Taking the second derivative of equation (3), we get

$$\ddot{\mathbf{\Omega}}_1(\mathbf{q}) = \dot{J}_{c1}(\mathbf{q})\dot{\mathbf{q}} + \mathbf{J}_{c1}(\mathbf{q})\ddot{\mathbf{q}} = 0. \tag{14}$$

Combining equations (13) and (14) together, the final EoM of subphase1 can be written as

$$\overline{\mathbf{M}_2(\mathbf{q})}\ddot{\mathbf{q}} + \overline{\mathbf{b}}_2(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \overline{\mathbf{g}_2(\mathbf{q})} = \overline{\mathbf{B}}_2\mathbf{u}, \tag{15}$$

where

$$\overline{\mathbf{M}}_2(\mathbf{q}) = \begin{bmatrix} \mathbf{J}_{c1null}^T\mathbf{M}_2 \\ \mathbf{J}_{c1} \end{bmatrix}_{(7\times7)},$$

$$\overline{\mathbf{b}}_2(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} \mathbf{J}_{c1null}^T\mathbf{b}_2 \\ \dot{J}_{c1} \end{bmatrix}_{(7\times7)},$$

$$\overline{\mathbf{g}}_2(\mathbf{q}) = \begin{bmatrix} \mathbf{J}_{c1null}^T\mathbf{g}_2 \\ 0_{2\times1} \end{bmatrix}_{(7\times1)},$$

$$\overline{\mathbf{B}}_2 = \begin{bmatrix} \mathbf{J}_{c1null}^T\mathbf{B} \\ 0_{2\times6} \end{bmatrix}_{(7\times6)}. \tag{16}$$

2.2.3. Double Support Subphase2.

The procedure of deriving the dynamic model of subphase2 is similar to subphase1, except we need to add one more constraint for q_1:

$$\mathbf{\Omega}_2 = \begin{bmatrix} \mathbf{\Omega}_1 \\ q_1 - \dfrac{3\pi}{2} \end{bmatrix} = 0. \tag{17}$$

Then, the same procedure of subphase1 is followed to compute EoM, but with caution when choosing the dependent ($\mathbf{q}_{de2}$) and independent ($\mathbf{q}_{in2}$) coordinates. If we choose like in subphase1, corresponding $\mathbf{J}_{c2de}$ will be rank deficient; i.e., no inverse matrix exists; thus, we need to adjust $\mathbf{q}_{de2}$ and $\mathbf{q}_{in2}$ into

$$\begin{cases} \mathbf{q}_{in2}, & [q_4 \ q_5 \ q_6 \ q_7]^T, \\ \mathbf{q}_{de2}, & [q_1 \ q_2 \ q_3]^T. \end{cases} \tag{18}$$

2.2.4. Transitions: SSP to DSP1, DSP1 to DSP2, and DSP2 to SSP.

When the robot changes from SSP to DSP1 (an abbreviation of DSP subphase1), the role of legs switches; i.e., the original swing leg becomes the support leg and the same to the original support leg. Also, the fixed coordinate changes to the new support leg heel. Here, we do not consider the impact effect since impact could induce the

destabilizing effects on the motion control of the robot. Thus, transition can be represented using the following equation:

$$\mathbf{q}_{\text{SSP}} = \mathbf{T}\mathbf{q}_{\text{DSP1}},$$

$$\dot{\mathbf{q}}_{\text{SSP}} = \mathbf{T}\dot{\mathbf{q}}_{\text{DSP1}},$$

$$\mathbf{T} = \begin{bmatrix} & & & & & & 1 \\ & & & & & 1 & \\ & 0 & & 1 & & & \\ & & & 1 & & & \\ & & 1 & & 0 & & \\ & 1 & & & & & \\ 1 & & & & & & \end{bmatrix}. \tag{19}$$

The transition from DSP1 to DSP2 (an abbreviation of DSP subphase2) and DSP2 to SSP, the generalized coordinates, and velocities are continuous.

3. Trajectory Optimization

In the prior section, we get the dynamic models of the biped robot in all phases. In this section, we use the direct collocation method to generate the optimal trajectory.

The direct collocation method converts the trajectory optimization problem into a nonlinear programming problem by approximating all the continuous functions in the original problem as polynomial splines. In this paper, we utilize the standard collocation method [22] with the Hermite–Simpson method. Our problem has three phases, so we need to use the multiphase method [18, 25]. It is like solving multiple single-phase problems simultaneously. The difference is there are linkage constraints between any two connected phases, hence combining into an integrated trajectory.

The following are the details of the optimal trajectory problem.

3.1. Optimization Objective.
In this paper, we aim to generate low energy costs and high stability trajectory. For the energy cost, one common criterion is the cost of transport (CoT), which is the energy cost of the average distance moved by the robot, but CoT is a hard cost function to optimize over since it usually derives discontinuous solutions [27]. Therefore, we use a simple but effective function—torque squared cost function:

$$\mathcal{L}_u = \int_0^T \left(\sum_{i=1}^6 u_i(t)^2 \right) dt, \tag{20}$$

where T is the time of trajectory.

This cost function usually leads to smooth solutions. The smooth solution has three advantages. The first one is that it is easy to approximate for piecewise polynomial spline and

thus decreases the error because of approximation. The second one is that it is tractable to apply to the real robot. The last one is that it can prevent the torque from becoming too large.

As for stability, we use ZMP stability criterion. The ZMP point can be derived by the following equation:

$$\text{ZMP} = \frac{\sum_{i=1}^n m_i \left(\ddot{y}_i + g \right) x_i - \sum_{i=1}^n m_i \ddot{x}_i y_i - \sum_{i=1}^n I_i \ddot{q}_i}{\sum_{i=1}^n m_i \left(\ddot{y}_i + g \right)}, \tag{21}$$

where $\ddot{x}_i$, $\ddot{y}_i$ are the horizontal and vertical accelerations of G_i, n is the number of links, and g is the gravity acceleration.

Then, we limit the actual ZMP point to our preplanned ZMP curve. Note we cannot directly use $|\text{ZMP} - \text{ZMP}_{\text{des}}|$ to measure the deviation from actual ZMP to the desired ZMP since the absolute function is not continuous. Therefore, we use the below hyperbolic tangent function to approximate absolute function:

$$\mathcal{L}_{\text{ZMP}} = (\text{ZMP} - \text{ZMP}_{\text{des}})\tanh\left(\frac{\text{ZMP} - \text{ZMP}_{\text{des}}}{\alpha} \right) \approx |\text{ZMP} - \text{ZMP}_{\text{des}}|, \tag{22}$$

where ZMP_{des} is the desired ZMP and α is a parameter adjusting the smooth of the function.

Combining these two objectives, the final objective is

$$\mathcal{L} = \mathcal{L}_u + k_{\text{zmp}}\mathcal{L}_{\text{ZMP}}, \tag{23}$$

where k_{zmp} is the coefficient to balance the effects of ZMP and energy cost since the actual $\mathcal{L}_{\text{ZMP}}$ is much smaller than $\mathcal{L}_u$. By chosen suitable k_{zmp} such that two costs are at the same order of magnitude; otherwise, the effects of ZMP would not arise.

3.2. Constraints.
A parade of constraints is required to generate a feasible trajectory. For different phases, different parts are constrained to produce an ideal solution. Usually, the constraints can be divided into path constraints and boundary constraints. Path constraints are the restriction along the trajectory, while boundary constraints only take effect at the beginning and the end. The constraints applied in this paper are explained as follows (path constraints first):

(1) One of the basic constraints is avoiding the hyperextension of the knee joint and preventing the foot contact with the shin:

$$q_2 - q_3 \le 0,$$

$$q_6 - q_5 \le 0, \tag{24}$$

$$q_7 - q_6 - \frac{3\pi}{2} \le 0.$$

(2) Contact force constraints. Since we assume there is no slippage, the contact force must be within the friction cone. Besides, the contact is unilateral. These constraints correspond to the first two modeling assumptions:

$$|F_x| - \mu F_y \le 0,$$
$$- F_y \le 0,$$
$$\left.\begin{array}{l}|\lambda_x| - \mu \lambda_y \le 0 \\ -\lambda_y \le 0\end{array}\right\} \text{DSP,} \quad (25)$$

where F_x and F_y denote forces in the support leg, which can be solved by Newton's second law. λ_x and λ_y are forces in the swing leg, which occur in the DSP. μ is the friction coefficient.

(3) During SSP, the swing leg should be constrained to above the ground, and setting the ground clearance can make the trajectory more desirable. Two options to do this: one way is to keep the swing foot above continuous-time function $f(t)$, and the other way is to prescribe continuous state function $f(x)$. Here, we choose time-based foot clearance:

$$f(t) = h\left(\frac{(t - 0)(t - T_{\text{SSP}})}{(T_{\text{SSP}}/2 - 0)(T_{\text{SSP}}/2 - T_{\text{SSP}})}\right)^2,$$
$$t \in [0, T_{\text{SSP}}], \quad (26)$$
$$f(t) - P_{7y}(t) \le 0,$$
$$f(t) - P_{71y}(t) \le 0,$$

where T_{SSP} means the duration of SSP, h is a set parameter of the highest height, and P_{71} represents the position of swing foot heel.

(4) At the beginning of SSP (toe-off), the Y-axis velocity of swing foot toe should be positive:

$$-\dot{P}_{7y} \le 0. \quad (27)$$

(5) At the beginning of SSP, swing foot toe should be at the ground, and at the end of SSP, swing foot heel should be at the ground:

$$P_{7y}(0) = 0,$$
$$P_{71}(T_{\text{SSP}}) = 0, \quad (28)$$

where 0 means the time is zero, i.e., beginning of SSP. T_{SSP} means the duration time of SSP, i.e., end of SSP.

(6) We set the robot walk at a given speed (1 m/s). The ankle horizontal displacement in a step is constrained to 0.8 m, and the cycle time is prescribed as $T = T_{\text{DSP}} + T_{\text{SSP}} = 0.8$ s. However, in this paper instead of the constraint of ankle movement, we constrain the distance from swing toe to support heel at the beginning of SSP and support toe to swing heel at the end of SSP. Besides, according to the knowledge that the percentage of DSP is about 20% of the time and the other 80% is SSP [2], the T_{DSP} and T_{SSP} are determined:

$$P_{0x}(0) - P_{7x}(0) = L_{\text{Toe2Heel}},$$
$$P_{71x}(T_{\text{SSP}}) - P_{00x}(T_{\text{SSP}}) = L_{\text{Toe2Heel}}, \quad (29)$$

where P_{00} represents the position of support foot toe.

(7) Since no impact, we constrain the swing heel velocity is zero just before contact. This constraint corresponds to the third modeling assumption:

$$\dot{P}_{71} = 0. \quad (30)$$

(8) Phase transition (linkage constraints): these constraints are corresponding to the transitions part of the last section. SSP to DSP1, there is the reset map (Equation (18)); DSP1 to DSP2, keep continuous; and DSP2 to SSP, keep periodic.

The parameters used in optimization are shown in Table 2.

The value of limits of T_{DSP1} is determined by trial and error.

4. Results

We solve trajectory optimization using ICLOCS2 [28] software with IPOPT [29] and linear solver MUMPS. We use random values between the lower limit and upper limit as the initial guess. For the two kinds of walking styles, we both solve 10 times from 10 random initial guesses. All optimizations are solved on a laptop computer (processor: 1.8 GHz quad-core Intel i7-8550U) with MATLAB R2019a. The average iterations and computation time are 593 iterations and 181 seconds for the first kind walking style. As for the second one, it takes much longer, and the average iterations and computation time are 1782 iterations and 546 seconds. Note here that we do not use any analytic information, and we only use IPOPT's numerical approximation. Stability [30] is not considered here. We give out the details of the generated trajectory for each walking style and the comparison of these two walking styles.

4.1. Results of the First Walking Style. The curves of each joint angle and joint velocity are given in Figures 3 and 4. The middle vertical line depicted in all figures of this section represents the dividing line, the left part is DSP, which also be divided into DSP1 and DSP2 by a middle vertical line in the second walking style, and the right is SSP. From these two figures, we can see the curves are smooth, and state switching from DSP to SSP is continuous. Since the ranges of q_1 and q_7 (angle of the foot) are different from others, they are above the others. Also, $\dot{q}_5$, $\dot{q}_6$, and $\dot{q}_7$ all experience large fluctuation during SSP since they swing from back to fore. We observe that the swing foot experiences the most fluctuation like a roller coaster, then the swing knee, and the last one is the swing hip. From the view of minimizing energy, it is reasonable first to adjust the minimum energy cost part. This gives us a hint of control method, and we should

TABLE 2: Parameters of optimization.

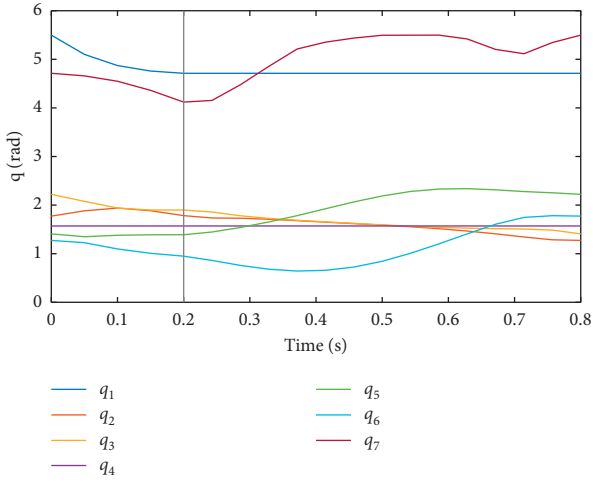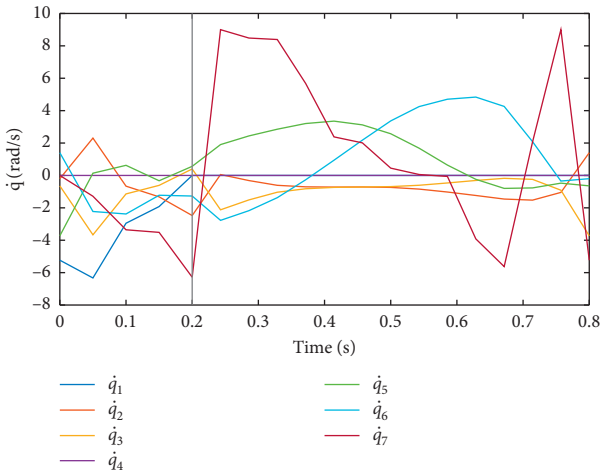| Symbol	Value
k_{ZMP}	$1e^4$
T_{DSP1}	[0.1 0.16] s
T_{DSP}	0.2 s
T_{SSP}	0.6 s
$T = T_{DSP} + T_{SSP}$	0.8 s
α	0.01
h	0.055 m
$L_{Toe2Heel}$	0.2495 m

FIGURE 3: The joint angle trajectories of gait1.

FIGURE 4: The joint velocity trajectories of gait1.

prioritize the control of the minimum energy cost part. The stick diagram is shown in Figure 5. During the medium of SSP, the knee joint is at the singularity position, and this leads to the cycloid curve of hip height, which is shown straightly in the next figure. The swing foot toe and heel are all above the ground clearance curve (green and yellow, respectively) during SSP. The hip height curve is shown in

Figure 6. It depicts a more straightforward view of the cycloid curve during SSP.

4.2. Results of the Second Walking Style. The optimal value of T_{DSP1} is 0.1161 s, which is about 58% of the whole DSP period. The joint angle and velocity trajectories of this walking style are plotted in Figures 7 and 8, respectively. They are also smooth and continuous when switching between different phases. Figure 9 illustrates the stick diagram. The hip height trajectory is depicted in Figure 10. The height of the hip joint is also a cycloid curve.

4.3. Comparison of Two Gaits. Comparing Figures 3 and 7, the joint angle trajectories are basically the same. As for the joint velocity (Figures 4 and 8), comparing to the first walking style, the scope of fluctuation in the second walking style when transferring from DSP to SSP is slightly bigger and at the end of SSP is much smaller. The energy cost of these two walking styles is different. Since both walking styles experience the same distance and time, we can directly use the equation below without dividing the distance to compare:

$$\text{power} = \sum_{k=1}^{N} |\tau_k \mathbf{q}_k|, \tag{31}$$

where N is the number of collocation points (20 in this paper) and τ is the generalized actuator torques in equation (1).

The energy cost of gait2 is 1040 joule, and it is slightly more than gait1 (915 joule). Although the second gait increases the cost of energy a little, it raises the stability. Figure 11 illustrates the trajectory of ZMP, the left one is gait1, and the right one is gait2. Both ZMP trajectories follow the desired ZMP and within the support polygon, which is the region between the black dotted line. In gait1, at the transferring moment from DSP to SSP, ZMP (depicts as the red circle) is at the ZMP upper limit of DSP, which leads to the trend of biped robot falling forward; also, it is at the ZMP lower limit of SSP, which leads to the trend of biped robot falling backward. In contrast, for gait2, ZMP at transferring moment is 0.053, which is one-third the distance between the ZMP lower limit and upper limit of SSP, so the stability margin is increased from 0 to 0.053 m, which prevents the falling down caused by small deviation, as we mentioned the benefit of this walking style at Section 2.2.

4.4. Effect of Weight between Energy Cost and Stability. We also investigate the influence of the weight between energy cost and stability on the result. We add a weight w to our objective function (Equation (21)):

$$\mathcal{L} = w\mathcal{L}_u + (1 - w)k_{zmp}\mathcal{L}_{ZMP}, \quad w \in [0, 1]. \tag{32}$$

We set five different weight values $w \in \{0, 0.25, 0.5, 0.75, 1\}$ and find that the energy cost decreases with the increase in weight in both gaits (Figure 12). Besides, when $w < 0.5$, the energy cost of two-phase gait is more than three-
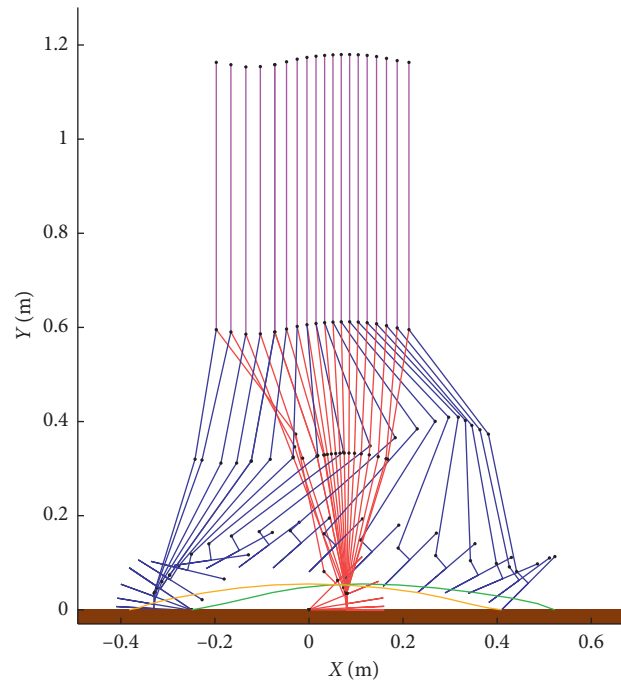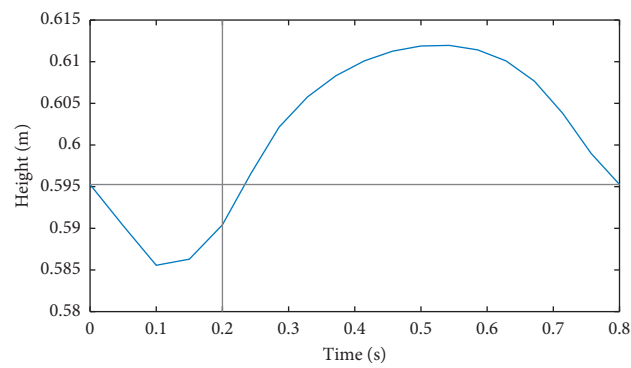
FIGURE 5: Stick diagram of gait1.

FIGURE 6: The hip height curve of gait1.

q_1
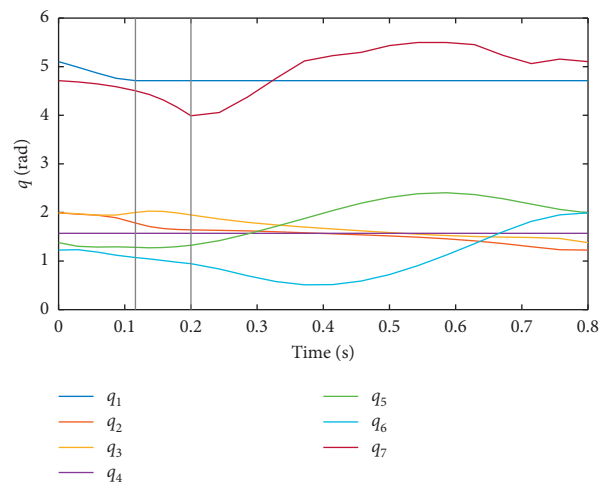q_2
q_3
q_4
q_5
q_6
q_7

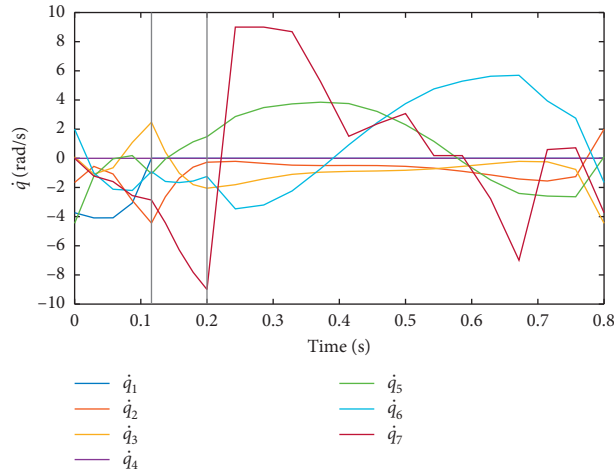FIGURE 7: The joint angle trajectories of gait2.

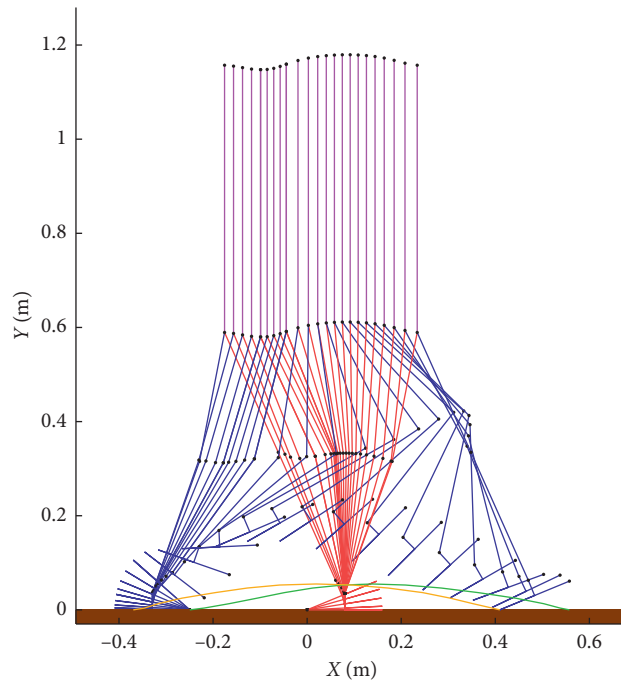FIGURE 8: The joint velocity trajectories of gait2.

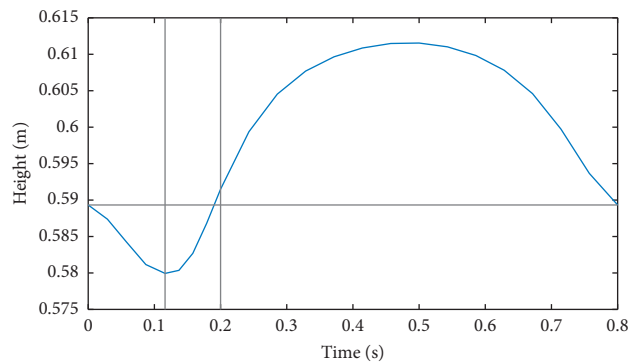FIGURE 9: Stick diagram of gait2.
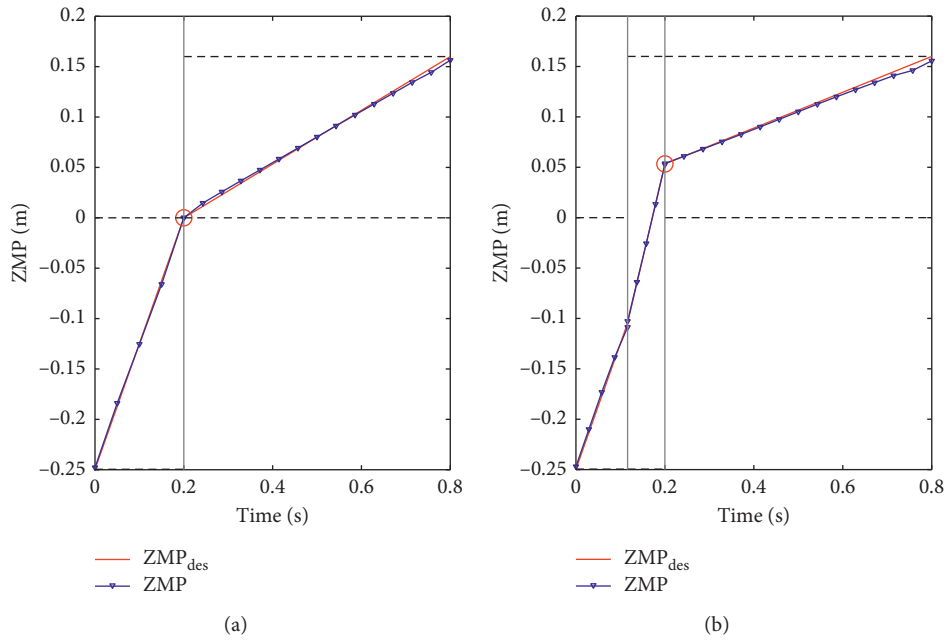

FIGURE 10: The hip height curve of gait2.

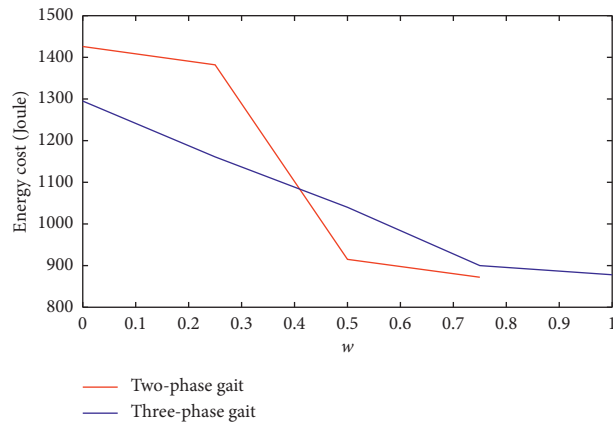FIGURE 11: The ZMP trajectories of two gaits. (a) The first gait. (b) The second gait.

FIGURE 12: The effect of weight on energy cost.

phase gait, but when $w \geq 0.5$, it is the opposite. Note that when $w = 1$, the two-phase gait could not get the optimal solution, so this point is not in Figure 12.

For the effect of weight on ZMP (Figure 13), when $w \leq 0.5$, although actual ZMP is slightly getting worse with the increase in weight, it still can follow the desired ZMP, but when $w > 0.5$, it is getting very bad. Note that when $w = 1$, the two-phase gait could not get the optimal solution, so there is only the ZMP curve of three-phase gait. Although when $w < 0.5$, the energy cost of gait1 is more than gait2, and it seems opposite to our conclusion, but at this time, the energy cost is more than $w = 0.5$, so our conclusion is still correct when considering energy cost and stability equally ($w = 0.5$).
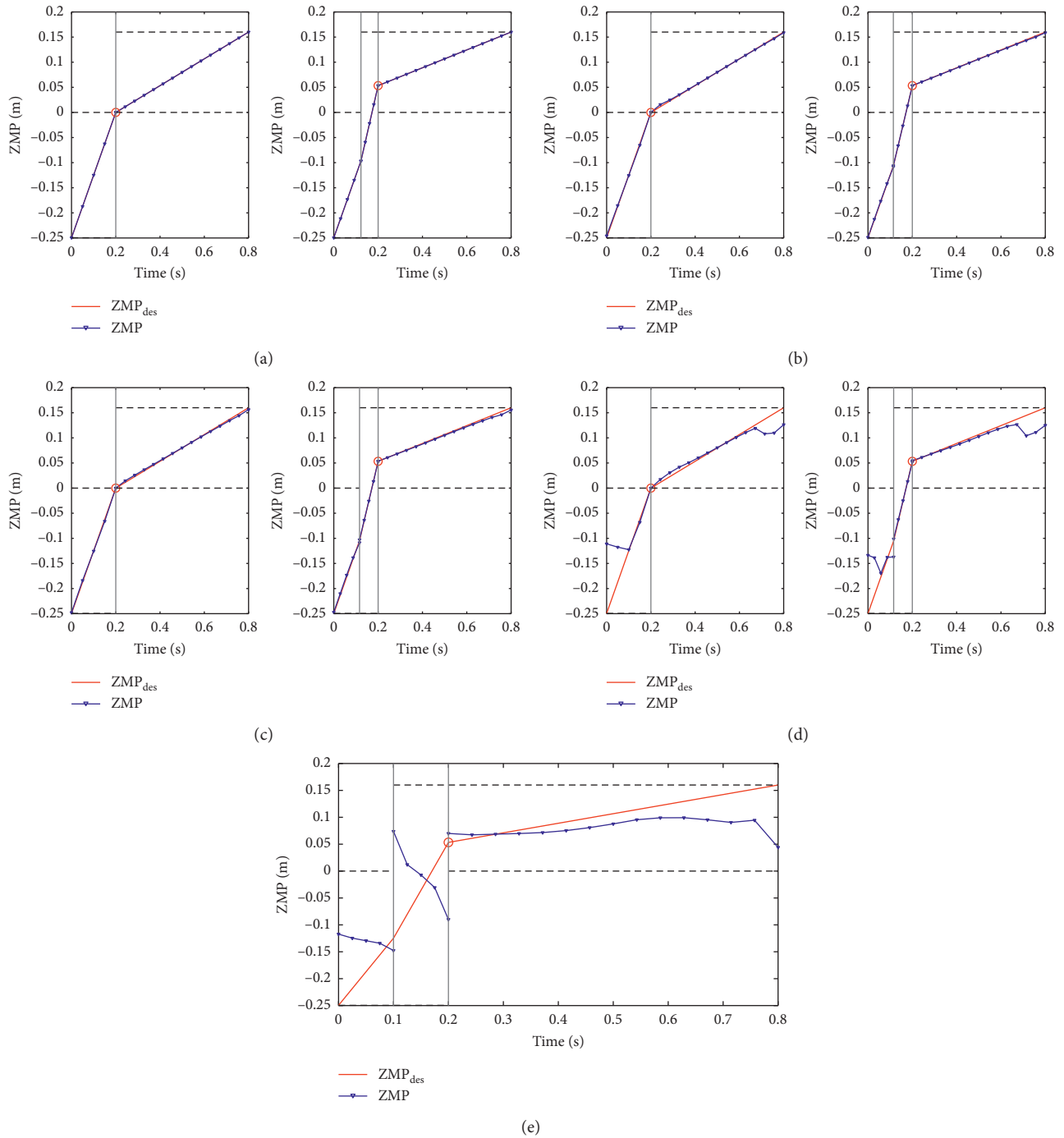
FIGURE 13: The effect of weight on ZMP. (a) $w = 0$ (left: gait1, right: gait2). (b) $w = 0.25$. (c) $w = 0.5$. (d) $w = 0.75$. (e) $w = 1$.

5. Conclusions and Discussion

Aiming to increase the stability of the biped robot when transferring from DSP to SSP, this paper divides DSP into two subphases. Then, we apply the direct collocation method, which is suitable for our problem and overcomes some limitations of parametric optimization and the DDP method, to generate the optimal multiphase trajectory toward stability and energy efficiency. In the end, we compare the three-phase trajectory with the traditional two phases.

The comparison confirms that multiphase gait can increase stability although it consumes slightly more energy. Besides, both trajectories show the human-like cycloid curve, which makes the gait look more natural. In the future, we will use more accurate and effective direct collocation method, like orthogonal collocation methods [31] instead of the standard collocation methods, which we use here. Besides, we will apply this trajectory to a physical robot and try to reduce computation time through a less complex dynamic model like centroidal momentum model. We will generate various

trajectories with different velocities and different foot ground clearances to build a library, which can make the robot more adaptive to different situations.

Data Availability

The data (MATLAB file) used to support the findings of this study are available from Shuai Heng upon request.

Conflicts of Interest

The authors declare that there are no conflicts of interest.

Authors' Contributions

Yubin Liu and Shuai Heng coorganized this work and wrote the manuscript draft. Xizhe Zang and Zhenkun Lin proposed some good ideas. Jie Zhao supervised the research and commented on the manuscript draft.

Acknowledgments

References

[1] M. H. Raibert, *Legged Robots that Balance*, MIT Press, Cambridge, MA, USA, 1986.

[2] H. F. N. Al-Shuka, F. Allmendinger, B. Corves, and W.-H. Zhu, "Modeling, stability and walking pattern generators of biped robots: a review," *Robotica*, vol. 32, no. 6, pp. 907–934, 2014.

[3] Y. Liu, X. Zang, S. Heng, Z. Lin, and J. Zhao, "Human-like walking with heel off and toe support for biped robot," *Applied Sciences*, vol. 7, no. 5, p. 499, 2017.

[4] D. Tlalolini, C. Chevallereau, and Y. Aoustin, "Comparison of different gaits with rotation of the feet for a planar biped," *Robotics and Autonomous Systems*, vol. 57, no. 4, pp. 371–383, 2009.

[5] C. Chevallereau, G. Bessonnet, G. Abba, and Y. Aoustin, *Bipedal Robots: Modeling, Design and Walking Synthesis*, Wiley/ISTE, New York, NY, USA, 2009.

[6] L. Jordan, M. J. Powell, and A. D. Ames, "Planar multi-contact bipedal walking using hybrid zero dynamics," in *Proceedings of 2014 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2582–2588, IEEE, Hong Kong, China, May 2014.

[7] F. Farzadpour, M. Danesh, and S. M. TorkLarki, "Development of multi-phase dynamic equations for a seven-link biped robot with improved foot rotation in the double support phase," *Proceedings of the Institution of Mechanical Engineers, Part C: Journal of Mechanical Engineering Science*, vol. 229, no. 1, pp. 3–17, 2015.

[8] M. Vukobratović and B. Borovac, "ZERO-MOMENT point—thirty five years of its life," *International Journal of Humanoid Robotics*, vol. 1, pp. 157–173, 2004.

[9] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris, *Feedback Control of Dynamic Bipedal Robot Locomotion*, CRC Press, Boca Raton, FL, USA, 2007.

[10] M. A. Elhosseini, A. Y. Haikal, M. Badawy, and N. Khashan, "Biped robot stability based on an A-C parametric Whale Optimization Algorithm," *Journal of Computational Science*, vol. 31, pp. 17–32, 2019.

[11] G. Bessonnet, S. Chessé, and P. Sardain, "Optimal gait synthesis of a seven-link planar biped," *The International Journal of Robotics Research*, vol. 23, no. 10-11, pp. 1059–1073, 2004.

[12] S. Feng, X. Xinjilefu, W. Huang, and C. G. Atkeson, *3D Walking Based on Online Optimization. 13th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, pp. 21–27, IEEE, Atlanta, GA, USA, 2013.

[13] R. Budhiraja, J. Carpentier, C. Mastalli, and N. Mansard, "Differential dynamic programming for multi-phase rigid contact dynamics," in *Proceedings of IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, IEEE, Beijing, China, November 2018.

[14] A. Pavlov, I. Shames, and C. Manzie, *Interior point differential dynamic programming*, 2020, https://arxiv.org/abs/2004.12710.

[15] J. T. Betts, *Practical Methods for Optimal Control and Estimation Using Nonlinear Programming*, Society for Industrial and Applied Mathematics (SIAM), Philadelphia, PA, USA, 2010.

[16] L. Wang, Z. Wang, and Z. Qian, "A meshfree method for inverse wave propagation using collocation and radial basis functions," *Computer Methods in Applied Mechanics and Engineering*, vol. 322, pp. 311–350, 2017.

[17] L. Wang, Z. Wang, Z. Qian, Y. Gao, and Y. Zhou, "Direct collocation method for identifying the initial conditions in the inverse wave problem using radial basis functions," *Inverse Problems in Science and Engineering*, vol. 26, no. 12, pp. 1695–1727, 2018.

[18] M. Kelly, "An introduction to trajectory optimization: how to do your own direct collocation," *SIAM Review*, vol. 59, no. 4, pp. 849–904, 2017.

[19] A. V Rao, "A survey of numerical methods for optimal control," *Advances in the Astronautical Sciences*, vol. 135, pp. 497–528, 2009.

[20] M. Hardt, K. Kreutz-Delgado, and J. W. Helton, "Optimal biped walking with a complete dynamical model," in *Proceedings of the 38th IEEE Conference on Decision and Control*, pp. 2999–3004, IEEE, Phoenix, AZ, USA, December 1999.

[21] W. Xi, Y. Yesilevskiy, and C. D. Remy, "Selecting gaits for economical locomotion of legged robots," *The International Journal of Robotics Research*, vol. 35, no. 9, pp. 1140–1154, 2016.

[22] A. Hereid, *Dynamic humanoid locomotion: hybrid zero dynamics based gait optimization via direct collocation methods*, Ph.D. Thesis, Georgia Institute of Technology, Atlanta, Geogia, 2016.

[23] W. L. Ma, Y. Or, and A. D. Ames, "Dynamic walking on slippery surfaces: demonstrating stable bipedal gaits with planned ground slippage," in *Proceedings of International conference on robotics and automation (ICRA)*, pp. 3705–3711, IEEE, Montreal, QC, Canada, May 2019.

[24] A. Hereid and A. D Ames, "FROST: Fast robot optimization and simulation toolkit," in *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 719–726, IEEE, Vancouver, BC, Canada, September 2017.

[25] C. Li, Y. Ding, and H.-W. Park, "Centroidal-momentum-based trajectory generation for legged locomotion," *Mechatronics*, vol. 68, Article ID 102364, 2020.

[26] X. Zang, J. Song, Y. Liu, and X. Zhou, "Design of a pneumatic variable stiffness ankle joint," in *Proceedings of the 2018 IEEE International Conference on Intelligence and Safety for Robotics (ISR)*, pp. 291–296, IEEE, Shenyang, China, August 2018.

[27] N. Talele and K. Byl, *Toward Efficient and Robust Biped Walking Optimization*, arXiv:1807.09905, 2018.

[28] Y. Nie, O. Faqir, and E. C Kerrigan, "ICLOCS2: solve your optimal control problems with less pain," in *Proceedings of the 6th IFAC Conference on Nonlinear Model Predictive Control*, IFAC Secretariat, Madison, Wisconsin (USA), August 2018.

[29] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical Programming*, vol. 106, no. 1, pp. 25–57, 2006.

[30] L. Wang and Z. Qian, "A meshfree stabilized collocation method (SCM) based on reproducing kernel approximation," *Computer Methods in Applied Mechanics and Engineering*, vol. 371, Article ID 113303, 2020.

[31] D. Garg, M. Patterson, W. W. Hager, A. V. Rao, D. A. Benson, and G. T. Huntington, "A unified framework for the numerical solution of optimal control problems using pseudospectral methods," *Automatica*, vol. 46, no. 11, pp. 1843–1851, 2010.