Robotica page 1 of 18. © Cambridge University Press 2019
doi:10.1017/S0263574719000237

Robust Bipedal Locomotion Based on a Hierarchical Control Structure

Jianwen Luo†‡, Yao Su¶, Lecheng Ruan¶, Ye Zhao§, Donghyun Kim‡‡, Luis Sentis‡‡‡ and Chenglong Fu‡*

†*Department of Mechanical and Energy Engineering, Southern University of Science and Technology, Shenzhen, China. E-mail: luojw@sustc.edu.cn*
‡*Department of Computer Science, Stanford University, Stanford, CA, USA*
¶*Mechanical and Aerospace Engineering Department, University of California, Los Angeles, CA, USA. E-mails: yaosu@g.ucla.edu, ruanlecheng@ucla.edu*
§*Woodruff School of Mechanical Engineering, Georgia Institute of Technology. E-mail: ye.zhao@me.gatech.edu*
‡‡*Mechanical Engineering Department, Massachusetts Institute of Technology, MA, USA. E-mail: robot.dhkim@gmail.com*
‡‡‡*Department of Aerospace Engineering and Engineering Mechanics, University of Texas at Austin, Austin, TX, USA. E-mail: lsentis@austin.utexas.edu*

(Accepted January 29, 2019)

SUMMARY

To improve biped locomotion's robustness to internal and external disturbances, this study proposes a hierarchical structure with three control levels. At the high level, a foothold sequence is generated so that the Center of Mass (CoM) trajectory tracks a planned path. The planning procedure is simplified by selecting the midpoint between two consecutive Center of Pressure (CoP) points as the feature point. At the middle level, a novel robust hybrid controller is devised to drive perturbed system states back to the nominal trajectory within finite cycles without chattering. The novelty lies in that the hybrid controller is not subject to linear CoM dynamic constraints. The hybrid controller consists of two sub-controllers: an oscillation controller and a smoothing controller. For the oscillation controller, the desired CoM height is specified as a sine-shaped function, avoiding a new attractive limit cycle. However, this controller results in the inevitable chattering because of discontinuities. A smoothing controller provides continuous properties and thus can inhibit the chattering problem, but has a smaller region of attraction compared with the oscillation controller. A hybrid controller merges the two controllers for a smooth transition. At the low level, the desired CoM motion is defined as tasks and embedded in a whole body operational space (WBOS) controller to compute the joint torques analytically. The novelty of the low-level controller lies in that within the WBOS framework, CoM motion is not subject to fixed CoM dynamics and thus can be generalized.

KEYWORDS: Robust locomotion; Generic point mass model; Whole body control; Operational space.

1. Introduction

Biped robots are morphologically similar to humans promising to adapt the former to everyday human environments. In those environments, bipeds need to effectively plan a feasible path to their target locations. To realize the locomotion control process, it is useful for the control architecture to be organized hierarchically where a high-level controller is subject to constraints determined from a

* Corresponding author. E-mail: fucl@sustc.edu.cn

low-level process closer to the physical layer.[1,2] As such, our study proposes a hierarchical control architecture to realize practical bipedal locomotion, the advantages of which lie in the following: (1) at the high level, a planning algorithm achieves longer time horizon planning extracting features for the path planning process; (2) at the middle level, a simplified Center of Mass (CoM) dynamic model of the locomotion process is used to analyze and synthesize walking controllers; and (3) at the low level, a whole body operational space (WBOS) controller computes joint torques to achieve the planned dynamic path.

At the high level, a sequence of feasible footholds is generated through path planning. The path planning considers the environment and the biped's dynamic constraints. Broadly speaking, there exist two families of approaches for footstep planning: discrete search and continuous optimization. Discrete search approaches have typically relied on generating a pre-computed action set; either represented as a set of possible displacements from one footstep to the next or a set of possible footholds in the environment. Actions are connected together to form a tree of possible footstep plans, which can be explored using existing discrete search methods like A^* or RRT. Action set approaches using pre-computed step displacements have been used in refs. [3–8]. However, applying A^* or other informed search methods to our problem would be complicated due to the difficulty of defining a good heuristic for partial footstep planning, that is, we cannot generally know how many additional footsteps a partial plan will need in order to reach the goal without actually searching for those steps.[9]

At the mid level, a robust tracking controller is used to make the robot's CoM track, the planned path given by the high-level planner. A fundamental problem is motion stability. With relatively large disturbances, the robot's limited sized supporting feet is insufficient for recovery, thus a new foot placement is required for stability. If we assume deterministic models, tools such as fixed-points, limit cycles, or controllability analysis can be used for locomotion synthesis. However, when there exist disturbances, such as model inaccuracies or unpredicted variations in terrain, tools from robust verification become useful to characterize the stability of the locomotion process. Robust stability analysis takes two interesting forms: worst-case scenario and stochastic analysis. To the best of our knowledge, worst-case analysis has received little attention to date in legged locomotion; however, approaches based on Lyapunov stability[10–13] can be adopted using the notion of a common Lyapunov function.[14] One challenge here is that nominal solutions – for example, the exact shape of the nominal limit cycle – are often parameter-dependent, that is, the system might still be stable with different parameters, but to a slightly different limit cycle.

At the low level, WBC models biped robots as rigid body dynamic systems with a floating base and physical constraints.[15] The robot is controlled to accomplish multiple tasks using operational space tasks through torque control commands. Robot contacts are modeled as constraints in an optimization problem[16,17] or via projection-based techniques.[18,19] An alternative way to WBC is employing quadratic programming,[20–25] which not only allows for inequality constraints to be represented but also endures slower computations. At the middle level, various simplified CoM dynamic models have been proposed such as the Linear Inverted Pendulum Model (LIPM),[26–29] for which the CoM height is constant and the CoM horizontal dynamics are linear. Another popular model is the spring-loaded inverted pendulum (SLIP).[30–33] SLIP captures compliance by using an elastic element within the supporting leg. This compliance is critical to achieve walking efficiency and contact safety. The prismatic inverted pendulum model (PIPM) is another practical point mass model which has been successfully implemented in a simulated three-dimensional point-foot biped robot.[34,35] PIPM considers the CoM's height variance and uses it to plan the robot's CoM motion in the horizontal plane.

Overall, our contributions in this paper are as follows: (1) At the high level, a midpoint-based method is proposed for locomotion path planning. Different from previous planning methods, midpoints between consecutive Center of Pressures (CoPs) are selected as feature points for generating a feasible and optimal CoM trajectory to track the path via A^* search. An optimization-based method is used to search an optimal sequence of footholds for turning at the intersection of consequent segmented lines planned by the A^* algorithm. (2) At the middle level, we propose and compare two novel robust hybrid trajectory controllers, which can bring a disturbed state back to its nominal trajectory on an extended phase space. Our hybrid controller newly combines an oscillation controller with a smoothing controller for effective performance. The proposed oscillation controller delivers good robust performance on CoM convergence but suffers from chattering. In contrast, the proposed smoothing controller has a smaller region of attraction, but suffers no chattering. Logically, the hybrid controller combines simultaneously the oscillation controller and the smoothing controller to achieve larger regions of attraction while avoiding chattering. (3) At the low level, a foot placement planner

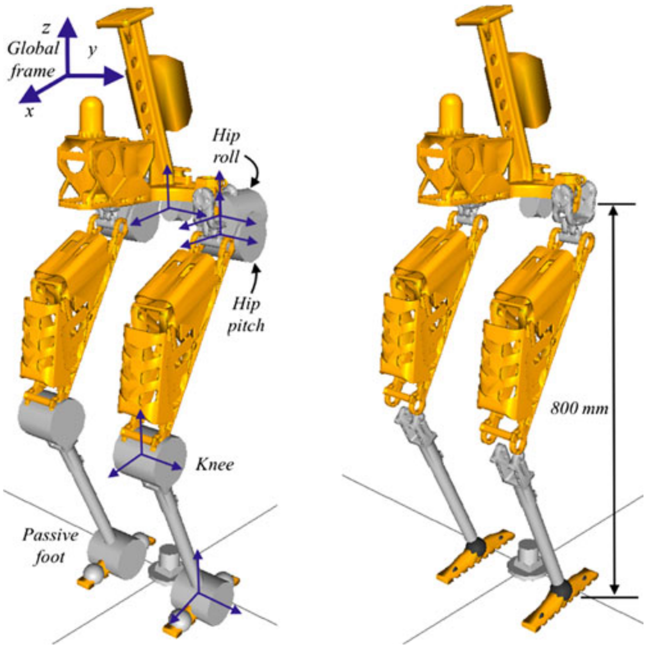


Fig. 1. Three-dimensional passive-foot biped robot. The left figure shows the robot coordinates and the right figure shows a clear linkage and joint kinematic view of the robot.

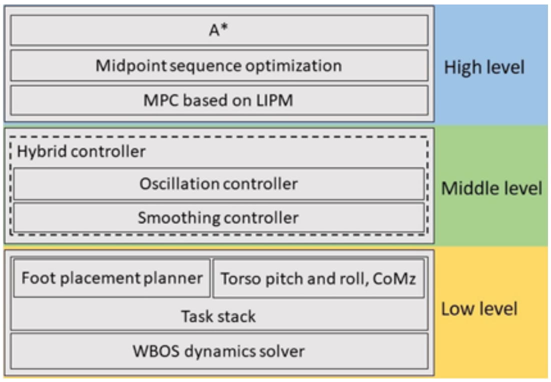


Fig. 2. Overall hierarchical control architecture for biped locomotion.

that is not strictly subject to a predefined point mass model is proposed and embedded in the WBOS framework. The CoM motion and the body posture are formulated as tasks in the WBOS framework. All tasks are glued together without task priority. Since the dynamics switch based on contact events, it is not required to specify a fixed swing time. This flexibility offers adaptability to uneven terrains.

The paper is organized as follows: Section 2 overviews the control structure of the biped locomotion process. Section 3 presents the path planning strategy. Section 4 focuses on analysis of the simplified CoM dynamics and the robust trajectory controller. The WBOS framework and the foot placement planner are discussed in Section 5. Conclusions are presented in Section 6.

2. Hierarchical Control Structure

Our simulated biped robot has six actuated DoFs, including one pitch DoF on each knee, one pitch DoF on each hip and one roll DoF on each hip. There is one passive pitch DoF on each ankle, which does not provide active ankle torque. Therefore, the simulated model represents a point-foot biped robot. The passive-foot provides yaw vertical friction torque with respect to the ground surface to counteract yaw momentum disturbances. The DoFs and coordinates of the biped robot are as shown in Fig. 1.

The hierarchical structure of the control framework is depicted in Fig. 2. At the high level, the A^* algorithm is used to plan a path from the starting location to the destination in a known 2D map.

The shortest feasible path that connects the starting point and destination is generated as a sequence of segmented lines. On each turning point, an arc is incorporated to smooth the transition path. Midpoints between two consecutive CoPs are selected as feature points. An optimization process is used to find the optimal sequence of midpoints and CoPs such that the CoM trajectory generated based on a model predictive controller (MPC) can track the planned path. MPC forms a closed loop which provides realtime converge of the CoM dynamics. However, robustness of MPC is weak to large disturbances. At the middle level, where MPC fails under larger disturbances, a hybrid controller guarantees the trajectory to be robust in the sense of N-cycle stability. The hybrid controller combines two sub-controllers: an oscillation controller and a smoothing controller. The oscillation controller attempts to drag the CoM state back to the nominal manifold in the extended phase space. The smoothing controller attempts to eliminate the chattering problem. At the low level, The CoM motion planner along with the foot placement planner are broken down into tasks and plugged into the WBOS framework to compute joint torques.

3. Path Planning

In this section, a path planing method based on MPC and A^* is discussed. MPC is incorporated to provide closed loop control of the CoM states with respect to a desire CoP. A^* generates an optimal path from a starting point to the destination. The path consists of segmented lines enhanced with an optimal sequence of footholds at the turning points.

3.1. CoM trajectory based on MPC

There is extensive work on biped locomotion via MPC.[36,37] For our method, LIPM dynamics are shown below:

$$\begin{cases} \dot{\hat{x}}_{k+1} = \begin{bmatrix} 1 & T & T^2/2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{bmatrix} \hat{x}_k + \begin{bmatrix} T^3/6 \\ T^2/2 \\ T \end{bmatrix} u_k \\ p_{x,k} = [1 \quad 0 \quad -z_c/g]\hat{x}_k \end{cases} \tag{1}$$

where $u_k = \dddot{x}_k$ and $\hat{x}_k = [x_k, \dot{x}_k, \ddot{x}_k]^T$, x_k is the discretized position of CoM at kth sample time step. $p_{x,k}$ is the CoP position planned prior for kth sample time step.

Given a sequence of reference CoPs $P^{ref}_{x,k+1} = \left[p^{ref}_{x,k+1}, p^{ref}_{x,k+2}, \ldots, p^{ref}_{x,k+N}\right]^T$ and the corresponding input is $U_k = [\dddot{x}_k, \dddot{x}_{k+1}, \ldots, \dddot{x}_{k+N}]^T$, through N steps' iteration from Eq. (1), an augmentation equation is derived as shown in Eq. (2):

$$\begin{cases} P_{x,k+1} = M_x\hat{x}_k + M_u U_k \\ M_x = \begin{bmatrix} 1 & T & T^2/2 - z_c/g \\ \vdots & \vdots & \vdots \\ 1 & NT & N^2T^2/2 - z_c/g \end{bmatrix}_{N\times 3} \\ M_u = \begin{bmatrix} T^3/6 - Tz_c/g & 0 & 0 \\ \vdots & \ddots & \vdots \\ (1+3N+3N^2)T^3/6 - Tz_c/g & \ldots & T^3/6 - Tz_c/g \end{bmatrix}_{N\times N} \end{cases} \tag{2}$$

where $P_{x,k+1} = [p_{x,k+1}, \ldots, p_{x,k+N}]^T$.

The length of $P_{x,k+1}$ is N, which also corresponds to the preview horizon. Eq. (2) can be formulated as a QP problem where the objective function is defined as follows:

$$\min_{U_k} = \frac{1}{2}Q\left(P_{x,i+1} - P^{ref}_{x,i+1}\right)^2 + \frac{1}{2}RU_k^2 \tag{3}$$

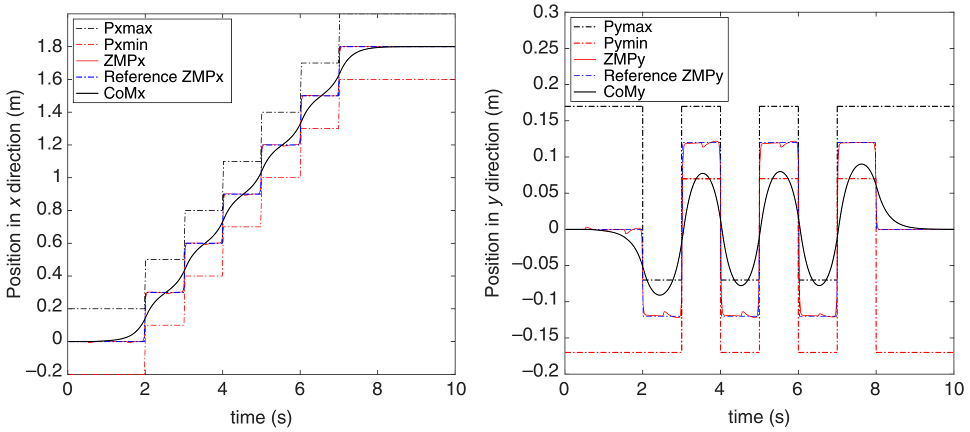


Fig. 3. This figure depicts the planned CoM trajectory based on the MPC process. The left and right figures show the horizontal and lateral trajectories of the robot's CoM, respectively.

where R and Q are weights shaping the tracking error and minimum effort, respectively. According to ref. [38] , the optimal solution with respect to the jerk takes the form below:

$$\dddot{x}_k = -e^T\left(\left(M_u^T M_u + \frac{R}{Q} I\right)^{-1} M_u^T\left(M_x \hat{x}_k - P^{ref}_{x,k+1}\right)\right) \tag{4}$$

where $e = [1, 0, \ldots, 0]^T_{1\times N}$ and I is the identity matrix of order N. The equation above provides an analytic optimal solution for the jerk, $\dddot{x}_k$, which also represents the input to the state Eq. (1) for the kth sampling time step. The CoM and CoP trajectories in the sagittal plane and frontal plane are shown in Fig. 3.

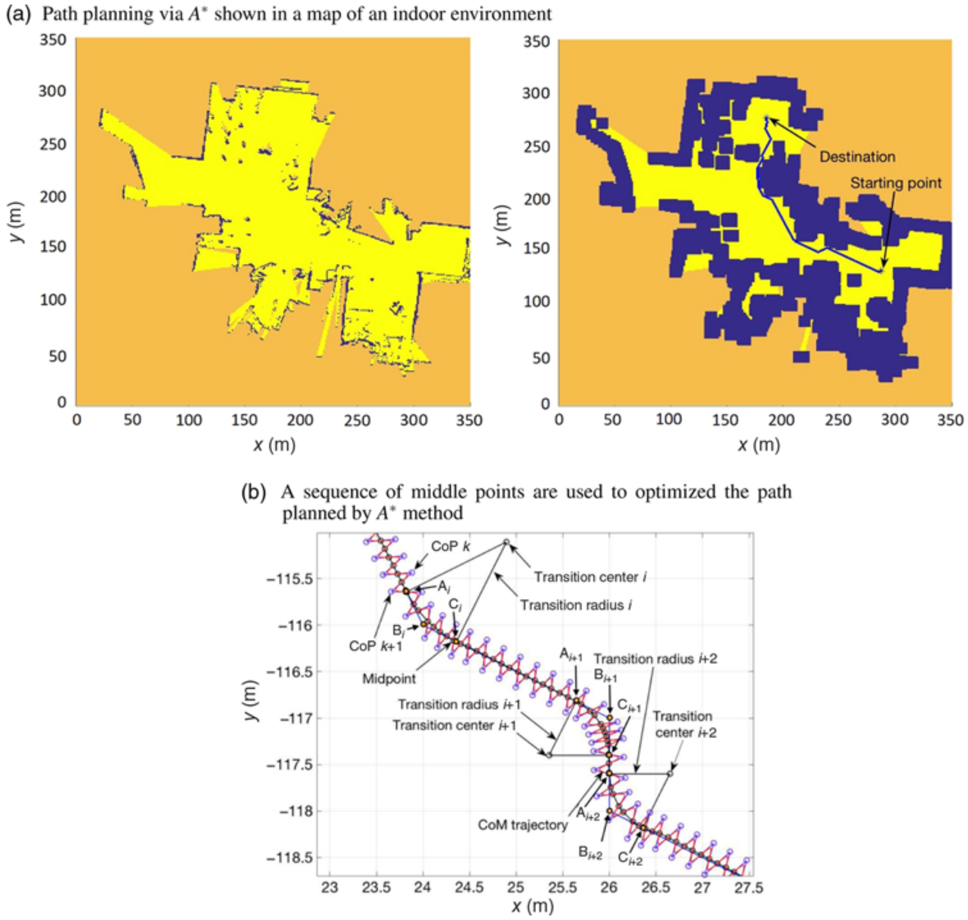
3.2. *Optimal footholds planning for turning*

We consider here a map of an indoor environment. We perform A^* search to generate the shortest feasible path from a starting location to the goal location in the map. Constraints in the environment are labeled as squares. The map and the path planned using A^* are shown in Fig. 4(a). Due to CoM dynamic constraints , the CoM evolves along a wave-shaped trajectory which is subject to the dynamics defined by Eq. (1). Our study relies on a feature point at the midpoint of stepping trajectories taken as the midpoint between the previous and the current CoPs. We define the midpoint and the corresponding midpoint sequence for the path planned by A^* as CoP[i] and Mid[i], $i \in \mathbf{N}$. Therefore:

$$\mathrm{Mid}[i] = \frac{\mathrm{CoP}[i] + \mathrm{CoP}[i+1]}{2} \tag{5}$$

A CoM trajectory generator that can track the path planned by A^* is devised based on these midpoints. The path is designed to maneuver in the map of Fig. 4(a). A transition planner at each turning point is used to generate a trajectory along an arc with a certain radius. As shown in Fig. 4(b), the length of the segmented lines A_iB_i, B_iC_i, $A_{i+1}B_{i+1}$, $B_{i+1}C_{i+1}$, $A_{i+2}B_{i+2}$, and $B_{i+2}C_{i+2}$ is designed given a fixed constant. From this, the arc radius can be computed. Along each straight segment, a constant step length is defined and a sequence of midpoints is generated along with the sequence of CoPs (footholds). Before the CoM approaches the next turning point, a reference arc is generated. A sequence of midpoints that can track the arc is found so that the CoM trajectory can turn smoothly along the arcs. A direction vector **w** and CoP vector **c** are defined to guide the walking trajectory. **w** is defined as the vector pointing from the previous midpoint to the current midpoint. **c** is defined as the vector pointing from the previous CoP to the current CoP. For the turning foothold sequence, we formulate an optimization problem that minimizes the angle between **w** and **c**:

$$\min\ \theta \qquad\qquad s.t. \quad \begin{cases} n = N \\ w_0 = \mathrm{Mid}[i_{start}] - \mathrm{Mid}[i_{start} - 1] \\ c_0 = \mathrm{CoP}[i_{start}] - \mathrm{CoP}[i_{start} - 1] \end{cases} \tag{6}$$


(a) Path planning via A^* shown in a map of an indoor environment

(b) A sequence of middle points are used to optimized the path planned by A^* method

Fig. 4. The map in (a) shows the map scanned from an indoor environment. The path planned uses the A^* algorithm from a given starting point to a goal point. Constraints are labeled in the map. (b) shows that midpoints are used as features to plan the sequence of CoPs and thus generate the CoM trajectory based on MPC. The red line is the trajectory of the CoM and the black curve is the path planned by A^* in (a).

where $\theta = \arcsin(< \mathbf{w}, \mathbf{c} > /||\mathbf{w}|| \cdot ||\mathbf{c})||)$. $< \mathbf{w}, \mathbf{c} >$ is the inner product of $\mathbf{w}$ and $\mathbf{c}$. $\mathbf{w} = w_n$ and $\mathbf{c} = \mathbf{c_n}$. N is the specified step number for turning.

4. Dynamic Control of a Robot's CoM

In this section, a point mass model and a nonlinear system are considered. A manifold and Lyapunov function are also defined. Results based on numeric computation for such system are demonstrated. Our study proposes two sliding mode controllers. One drives the disturbed states to converge quickly to a nominal trajectory. The other one guarantees that the disturbed states converge to the nominal trajectory without chattering. A hybrid controller combines the two controllers to realize a robust policy.

4.1. Point mass model with angular momentum

Simplified point mass models, such as IP, SLIP, LIPM, and PIPM, are all subject to the following CoM dynamics:

$$(r_{\text{CoP}} - r_{\text{CoM}}) \times f_{\text{GRF}} = -\tau_{\text{CoM}} \tag{7}$$

where τ_{CoM} can be considered as a disturbance on the CoM's moments. Generally speaking, if the angular momentum is considered, we can use the centroidal moment pivot (CMP) criteria, from which the ground reaction force (GRF) lies along the line between the CMP and the CoM. Usually angular momentum is quite small and can be considered as a uncertainty or disturbance. Therefore, if we ignore the angular momentum of the biped robot or eliminate the disturbance term, Eq. (7) can be written as Eq. (8):

$$(r_{\mathrm{CoP}} - r_{\mathrm{CoM}}) \times f_{\mathrm{GRF}} = 0 \tag{8}$$

where the position of the CoM, r_{CoM}, and the position of supporting foot's CoP, r_{CoP}, are $(x, y, z)^{\mathrm{T}}$ and $(x_{\mathrm{CoP}}, y_{\mathrm{CoP}}, z_{\mathrm{CoP}})^{\mathrm{T}}$ respectively. f_{GRF} is the GRF.

When there is no external forces, the excepted GRF and gravity force, f_{GRF}, are given by Eq. (9):

$$f_{\mathrm{GRF}} = \begin{bmatrix} m\ddot{x} \\ m\ddot{y} \\ m\ddot{z} + mg \end{bmatrix} \tag{9}$$

Combining Eqs. (8) and (9) yields the following equation:

$$\begin{bmatrix} 0 & -m\ddot{z} - mg & m\ddot{y} \\ m\ddot{z} + mg & 0 & -m\ddot{x} \\ -m\ddot{y} & m\ddot{x} & 0 \end{bmatrix} \begin{bmatrix} x - x_{\mathrm{CoP}} \\ y - y_{\mathrm{CoP}} \\ z - z_{\mathrm{CoP}} \end{bmatrix} = 0 \tag{10}$$

From the first two rows of the above equation, we get the following:

$$\ddot{x} = \frac{\ddot{z} + g}{z}(x - x_{\mathrm{CoP}}) \tag{11}$$

$$\ddot{y} = \frac{\ddot{z} + g}{z}(y - y_{\mathrm{CoP}}) \tag{12}$$

Let $\lambda = \lambda(\ddot{z}, z)$, then we have

$$\begin{cases} \ddot{x} = \lambda(\ddot{z}, z)(x - x_{\mathrm{CoP}}) \\ \ddot{y} = \lambda(\ddot{z}, z)(y - y_{\mathrm{CoP}}) \\ \lambda(\ddot{z}, z) = \dfrac{\ddot{z} + g}{z} \end{cases} \tag{13}$$

4.2. Linearized point mass model template

We use LIPM as a template model. If λ is constant, the dynamics of Eq. (13) is linearized. Note that $f_{\mathrm{GRFz}} = m\ddot{z} + mg$ is a unilateral ground force, therefore, λ should always be greater than zero, that is:

$$\begin{cases} \ddot{z} + g \geq 0 \\ z > 0 \end{cases} \tag{14}$$

The analytical solution in the lateral plane is

$$y(t) = (y(0) - y_{\mathrm{CoP}}) \cosh\left(\sqrt{\lambda} t\right) + \frac{\dot{y}(0)}{\sqrt{\lambda}} \sinh\left(\sqrt{\lambda} t\right) + y_{\mathrm{CoP}} \tag{15}$$

The derivative of $y(t)$ is

$$\dot{y}(t) = \sqrt{\lambda}(y(0) - y_{\mathrm{CoP}}) \sinh\left(\sqrt{\lambda} t\right) + \dot{y}(0) \cosh\left(\sqrt{\lambda} t\right) \tag{16}$$

Note that $\cosh^2\left(\sqrt{\lambda} t\right) - \sinh^2\left(\sqrt{\lambda} t\right) = 1$ and combining Eqs. (15) and (16), we get:

$$(y - y_{\mathrm{CoP}})^2 - \frac{\dot{y}^2}{\lambda} = (y_0 - y_{\mathrm{CoP}})^2 - \frac{\dot{y}_0^2}{\lambda} \tag{17}$$

For convenience, $y(t)$ and $\dot{y}(t)$ are replaced by y and $\dot{y}$, respectively, $y(0)$ and $\dot{y}(0)$ are replaced by y_0 and $\dot{y}_0$, respectively.

4.3. Robust hybrid controller design

In this subsection, a hybrid controller consisting of two sub-controllers is proposed. One controller is dubbed the oscillation controller, which can effectively drag the disturbed state back to the nominal trajectory. The second controller is dubbed the smoothing controller, which prevents chattering. These two sub-controllers rely on the sliding mode method.

4.3.1. Manifold analysis. When $\ddot{z}=0$ and z is constant, $\lambda=\lambda_0$, and the trajectory $(y(t), \dot{y}(t))$ will evolve along a hyperbolic manifold. Given y_{CoP}, we define the manifold s as follows in Eq. (18):

$$s(y, \dot{y}, \lambda)=(y-y_{\text{CoP}})^2-\frac{\dot{y}^2}{\lambda}-(y_0-y_{\text{CoP}})^2+\frac{\dot{y}_0^2}{\lambda} \tag{18}$$

For any state $(y, \dot{y}, \lambda)$ in the manifold s equals to 0. Thus, s defines the error between the current state and the nominal manifold. Take the derivative of $s(y, \dot{y}, \lambda)$, we get:

$$\dot{s}=2\dot{y}(y-y_{\text{CoP}})-\frac{2}{\lambda}\dot{y}\ddot{y}+\frac{\dot{\lambda}}{\lambda^2}\dot{y}^2-\frac{\dot{\lambda}}{\lambda^2}\dot{y}_0^2 \tag{19}$$

Note that $\ddot{y}=\lambda(y-y_{\text{CoP}})$. The equation above simplifies to:

$$\dot{s}=\frac{\dot{\lambda}}{\lambda^2}\left(\dot{y}^2-\dot{y}_0^2\right) \tag{20}$$

Define the Lyapunov function $V=s^2/2$, then $V\geq 0$ and

$$\dot{V}=s\dot{s}=s\frac{\dot{\lambda}}{\lambda^2}\left(\dot{y}^2-\dot{y}_0^2\right). \tag{21}$$

If $\dot{V}\leq 0$, V will vanish eventually. In an extended phase space defined as $(y, \dot{y}, \lambda)$, the state will converge to the manifold defined by Eq. (18). And V can be shown to be bounded within a maximum tube radius $s_{\max}^2/2$. The CoM height should subject to the condition defined below:

$$\begin{cases}\ddot{z}+g\geq 0\\ |z-z_n|\leq \boldsymbol{\varepsilon}_{\max}\\ \dot{V}<0\end{cases} \tag{22}$$

The manifold s defined in Eq. (18) is a function of y, $\dot{y}$, and λ. Define the space $\Omega=\{(y, \dot{y}, \lambda)|y\in R, \dot{y}, \lambda\in R^+\}$. λ depends on the CoM height dynamics. Even if the CoM height and λ are dynamically changing, we want the state $\xi=(y, \dot{y}, \lambda)\in\Omega$ to converge to the nominal manifold. Figure 5 demonstrates the subset of the manifold when one variable is ξ is fixed.

4.3.2. Oscillation controller. We propose an oscillation controller such that the desired trajectory of the CoM follows a sinusoidal function:

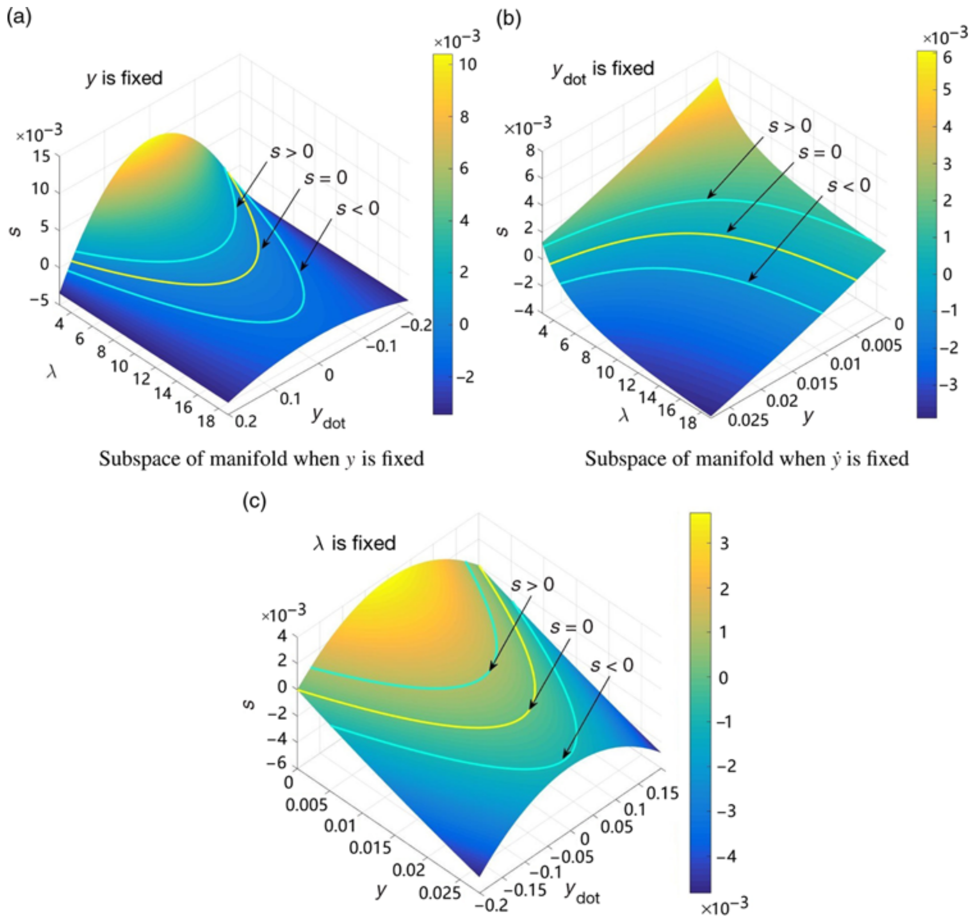
$$z=A\sin(\omega t+\phi)+z_n \tag{23}$$

where z_n is a nominal height of the CoM. The velocity and acceleration of z are:

$$\begin{cases}\dot{z}=A\omega\cos(\omega t+\phi)\\ \ddot{z}=-A\omega^2\sin(\omega t+\phi)\end{cases} \tag{24}$$

Note that $\ddot{z}=-\omega^2(z-z_n)$. According to Eq. (13), we get:

$$\begin{cases}\lambda=\dfrac{g-A\omega^2\sin(\omega t+\phi)}{A\sin(\omega t+\phi)+z_n}\\ \dot{\lambda}=-\dot{z}\dfrac{g+\omega^2 z_n}{z^2}\end{cases} \tag{25}$$



Subspace of manifold when y is fixed

Subspace of manifold when $\dot{y}$ is fixed

Subspace of manifold when λ is fixed

Fig. 5. Subspaces of manifold when y, $\dot{y}$, and λ are fixed, respectively. The yellow line corresponds to the nominal manifold.

Plug the equations above into Eq. (20), we get the derivative of the manifold in terms of the oscillation controller:

$$\dot{s} = \frac{-\dot{z}\left(\dot{y}^2 - \dot{y}_0^2\right)(g + \omega^2 z_n)}{z^2\lambda^2} \tag{26}$$

According to Eq. 21, $\dot{V} = \dot{s}s$. Then $\dot{V}$ becomes:

$$\dot{V} = -\frac{s\dot{z}\left(\dot{y}^2 - \dot{y}_0^2\right)(g + \omega^2 z_n)}{z^2\lambda^2} \tag{27}$$

Plug $\dot{z} = A\omega\cos(\omega t + \phi)$ in the above equation, we get:

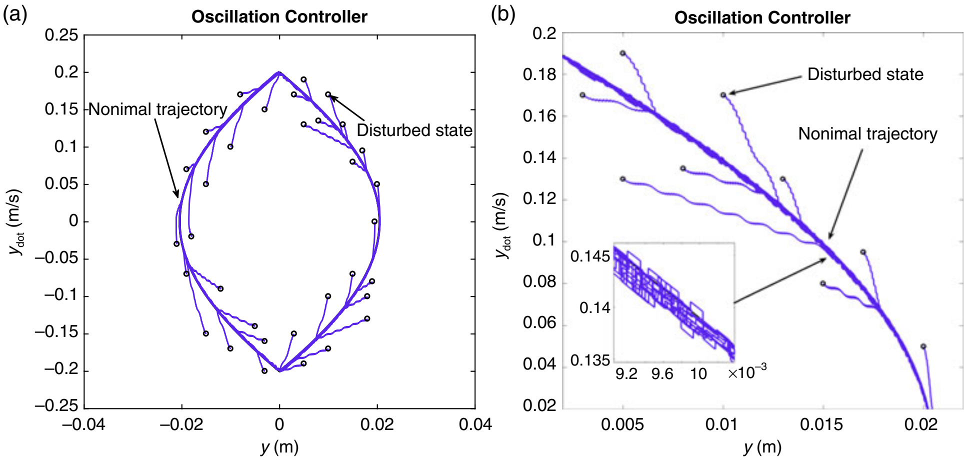
$$\dot{V} = s\frac{A\omega\cos(\omega t + \phi)\left(\dot{y}_0^2 - \dot{y}^2\right)(g + \omega^2 z_n)}{z^2\lambda^2} \tag{28}$$

Let $\dot{V} \leq 0$, considering that $\dot{y}^2 < \dot{y}_0^2$, $\omega > 0$, we get:

$$s\,A\cos(\omega t + \phi) \leq -|\varepsilon_{osc}| \tag{29}$$

where ε_{osc} is a bounded value to be specified. As long as the condition defined in the above equation is satisfied, there exists a finite positive value ρ such that the inequality below can be guaranteed:

$$\dot{V} \leq -|\rho| \tag{30}$$



Disturbed state converging to a nominal trajectory

Chattering issue caused by oscillation controller

Fig. 6. Sampled disturbed states converging to a nominal trajectory. (a) shows some disturbed states deviating from the nominal trajectory evolving back to the nominal trajectory under the effect of the oscillation controller. (b) shows that when the states converge, there is chattering effects along the trajectory.

Therefore, Eq. 29 represents the condition by which we can guarantee that s converges to the nominal manifold. When the disturbed state is on one side of the manifold, the distanced defined in Eq. 18 will be non-zero. During numerical computation, the sign of the manifold will stay unchanged and the equalities below hold:

$$\begin{cases} z = \mathrm{sign(s)}A\sin(\omega t + \phi) + z_n \\ \dot z = \mathrm{sign(s)}A\omega\cos(\omega t + \phi) \\ \ddot z = -\mathrm{sign(s)}A\omega^2\sin(\omega t + \phi) \end{cases} \tag{31}$$

where A and ϕ are constant terms. Subject to the first constraint of Eq. 22, we get:

$$g \geq \max\{A\omega^2\sin(\omega t + \phi)\} = |A|\omega^2 \tag{32}$$

The algorithm for the oscillation controller is shown in Algorithm 1. T is the period for the LIPM model. ξ is a factor that determines the maximum period time allowed in the controller. ε is the margin used to jump from the left half period to the right half period. N is the ratio between the control period and the computing period. t is the control period. Fig. 6 shows the states converge under Algorithm 1 and the chattering effects.

4.3.3. Smoothing controller. The oscillation controller can drag the disturbed states back to the nominal trajectory. However, due to the effect of the sign function, chattering around the nominal trajectory is inevitable. To smooth the trajectory, a smoothing controller is proposed, which can guarantee asymptotic convergence. This controller defines a new manifold σ as shown in Eq. (33):

$$\sigma = \alpha_1 s^2 + \alpha_2(\lambda - \lambda_n)^2 \tag{33}$$

where α_1 and α_2 are nonnegative coefficients, and hence σ is nonnegative. If s converges to zero, the states of the CoM will return to the nominal trajectory. If $\lambda - \lambda_n$ converges to zero, the CoM height will be stabilized within a small region around the desired height. The derivative of σ is

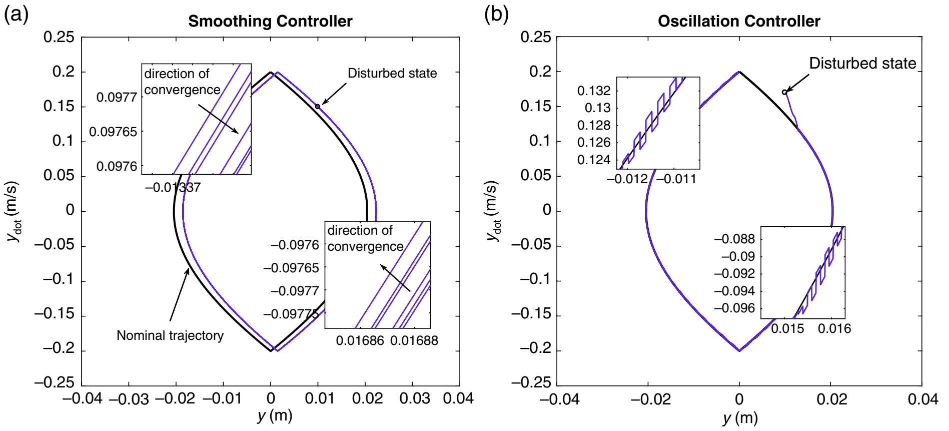
$$\dot\sigma = 2\alpha_1 s\dot s + 2\alpha_2(\lambda - \lambda_n)\dot\lambda \tag{34}$$

Define a Lyapunov function $V_\sigma = \sigma^2/2$. It's derivative is

$$\dot V_\sigma = \sigma\dot\sigma = \sigma\left[2\alpha_1 s\frac{\dot\lambda}{\lambda^2}\left(\dot y^2 - \dot y_0^2\right) + 2\alpha_2(\lambda - \lambda_n)\dot\lambda\right] \tag{35}$$

Algorithm 1 Oscillation Controller

```
while $t_k \le \xi T$ and $\dot{y}_k \ge -y_0 + \varepsilon$ and $\dot{y}_k \le y_0$ do
    $i \Leftarrow 0$
    while $i \le N$ do
        $s_{k,i} \Leftarrow (y_{k,i} - y_{CoP})^2 - \frac{\dot{y}_{k,i}^2}{\lambda_{a,k}} - (y_0 - y_{CoP})^2 + \frac{\dot{y}_0^2}{\lambda_{a,k}}$
        $\ddot{y}_{k,i} \Leftarrow \lambda_{a,k}(y_{k,i} - y_{CoP})$
        $\dot{y}_{k,i+1} \Leftarrow \dot{y}_{k,i} + \ddot{y}_{k,i} dt/N$
        $y_{k,i+1} \Leftarrow y_{k,i} + \dot{y}_{k,i} dt/N$
        $i \Leftarrow i + 1$
    end while
    $z_{k+1} \Leftarrow sign(s_{k,N}) A \sin(\omega t_k + \phi) + z_n$
    $\dot{z}_{k+1} \Leftarrow sign(s_{k,N}) A\omega \cos(\omega t_k + \phi)$
    $\ddot{z}_{k+1} \Leftarrow -sign(s_{k,N}) A\omega^2 \sin(\omega t_k + \phi)$
    $\lambda_{a,k+1} \Leftarrow (\ddot{z}_{k+1} + g)/z_{k+1}$
    $t_k \Leftarrow t_k + dt$
    $k \Leftarrow k + 1$
end while
```



The disturbed state slowly converges to the nominal trajectory using the smoothing controller

The disturbed state can not avoid chattering due to the oscillation controller

Fig. 7. The disturbed state converges asymptotically to the nominal trajectory when using the smoothing controller but can not reach the nominal trajectory. Comparatively, the oscillation controller suffers from chattering.

Let $\dot{V}_\sigma < 0$. We get:

$$\sigma\dot{\lambda}\left[\alpha_1 \frac{s\left(\dot{y}^2 - \dot{y}_0^2\right)}{\lambda^2} + \alpha_2(\lambda - \lambda_n)\right] < 0 \tag{36}$$

$$\dot{\lambda} = -\beta \tanh\left(\alpha_1 \frac{s\left(\dot{y}^2 - \dot{y}_0^2\right)}{\lambda^2} + \alpha_2(\lambda - \lambda_n)\right) \tag{37}$$

Looking at Fig. 7, since in the manifold defined by the smoothing controller, s and λ are designed to return to zero and to λ_n respectively, they could end up in another equilibrium orbit, meaning that λ could converge to a different $\lambda_n^{'} \neq \lambda_n$ and the disturbed state will continue in the trajectory. Compared to the oscillation controller, the smoothing controller avoids the chattering problem but fails to assure the convergence to the nominal trajectory. When the disturbed states are far away from the nominal trajectory, the smoothing controller is not capable of pulling them back. Figure 8 shows σ, for which the state evolves along the nominal trajectory of Fig. 7(a). σ is almost zero, which guarantees λ to stay close to λ_n.

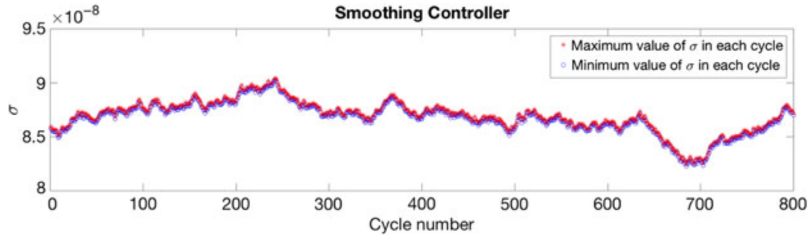


Fig. 8. σ changes very little in the smoothing controller, which implies that σ is almost zero, s is also almost zero, and λ tends to λ_n.

The computational algorithm is shown in Algorithm 2. T is the period for the LIPM model. ξ is a factor defining the maximum period allowed for the controller. ε is the margin used to jump from the left half period to the right half period. N is the ratio between the control period and the computing period. t is the control period. β α_1 α_2 are shown in Eqs. (36) and (37).

Algorithm 2 Smoothing Controller

```
while $t_k \leq \xi T$  and  $\dot{y}_k \geq -y_0 + \boldsymbol{\varepsilon}$  and  $\dot{y}_k \leq y_0$ do
    $i \Leftarrow 0$
    while $i \leq N$ do
        $s_{k,i} \Leftarrow (y_{k,i} - y_{CoP})^2 - \frac{\dot{y}_{k,i}^2}{\lambda_{b,k}} - (y_0 - y_{CoP})^2 + \frac{\dot{y}_0^2}{\lambda_{b,k}}$
        $\ddot{y}_{k,i} \Leftarrow \lambda_{b,k}(y_{k,i} - y_{CoP})$
        $\dot{y}_{k,i+1} \Leftarrow \dot{y}_{k,i} + \ddot{y}_{k,i}dt/N$
        $y_{k,i+1} \Leftarrow y_{k,i} + \dot{y}_{k,i}dt/N$
        $i \Leftarrow i + 1$
    end while
    $\dot{\lambda}_{b,k} \Leftarrow -\beta \tanh\left(\alpha_1 \frac{s_{k,N}\left(\dot{y}_{k,N}^2 - \dot{y}_0^2\right)}{\lambda_{b,k}^2} + \alpha_2(\lambda_{b,k} - \lambda_n)\right)$
    $\lambda_{b,k+1} \Leftarrow \lambda_{b,k} + \dot{\lambda}_{b,k}dt$
    $z_{k+1} \Leftarrow z_k + \dot{z}_k dt$
    $\dot{z}_{k+1} \Leftarrow \dot{z}_k + \ddot{z}_k dt$
    $\ddot{z}_{k+1} \Leftarrow z_{k+1}\lambda_{b,k+1} - g$
    $t_{k+1} \Leftarrow t_k + dt$
    $k \Leftarrow k + 1$
end while
```

4.3.4. Hybrid controller. The smoothing controller could avoid the chattering problem. However, it can only drag neighboring states back to the nominal trajectory while the oscillation controller, in comparison, is able to steer disturbed states far from the nominal trajectory. To inherit the merits of these two controllers, a hybrid controller is proposed. Intuitively, the oscillation controller and the smoothing controller are merged via weights. Through adjusting the weights, the dominance of each controller will gradually transition from one to the other. The weighted controller is defined as follows:

$$\lambda = \eta\lambda_a + (1 - \eta)\lambda_b \tag{38}$$

where λ_a and λ_b are given by the oscillation and the smoothing controllers respectively and shown in Eq. (39). The coefficient η is the weight used to adjust the transition between the two controllers. In this study η is set to have a linear relationship with the distance between the disturbed state and the nominal manifold, $\eta \in [0, 1]$. When the disturbed state is far away from the nominal trajectory, a larger η makes the oscillation controller dominate, speeding convergence. When the disturbed state comes near the nominal trajectory, η is tuned to be smaller so that the smoothing controller dominates. The combined effect stabilizes the system and avoids the chattering problem:

$$\begin{cases} \lambda_a = -\dot{z}\frac{g+\omega^2 z_n}{z^2} \\ \lambda_b = \int_{t_0}^{t} -\beta \tanh\left(\alpha_1 \frac{s\left(\dot{y}^2 - \dot{y}_0^2\right)}{\lambda^2} + \alpha_2(\lambda - \lambda_n)\right) \mathrm{d}t \end{cases} \tag{39}$$

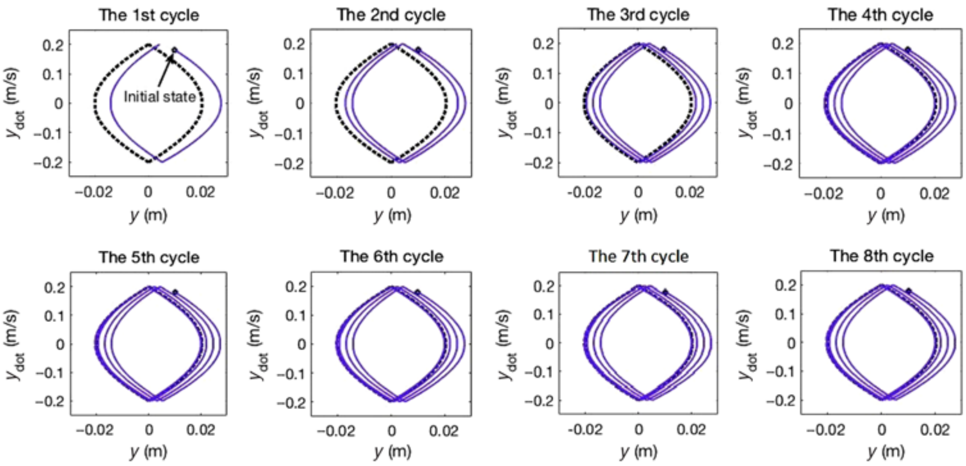


Fig. 9. An initial state that deviates from the nominal trajectory, asymptotically converges to the nominal trajectory by the aid of our proposed hybrid controller. The black dotted curve is the nominal trajectory.

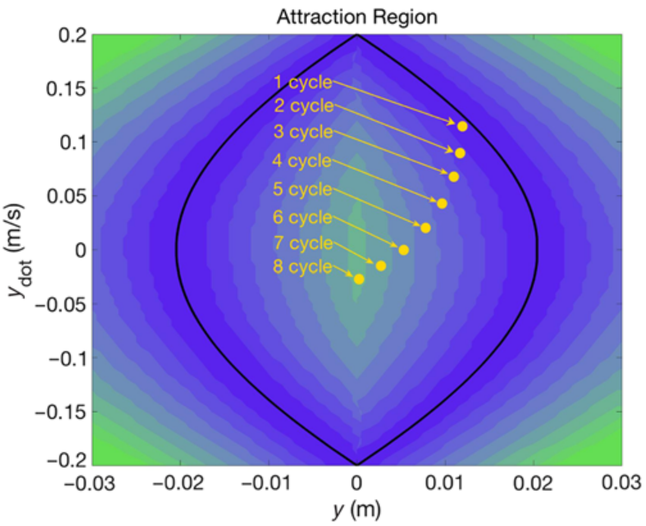


Fig. 10. Attraction region in the phase space. The black dotted curve is the nominal trajectory. Similar colors are used to label regions within which all the states can converge to the nominal trajectory within a number of cycles.

We define a $N-$region in the phase space, for which any initial state ξ can be dragged back to a small tube under the hybrid controller. The tube wraps around the nominal trajectory. Figure 9 shows an example of an initial state that asymptotically converges to the nominal trajectory with the hybrid controller. From the 4th cycle, the state basically stays on a trajectory that almost coincides with the nominal one. Figure 10 shows the $N-$ region decomposition. The same color mapping represents the number of cycles to converge to the nominal trajectory.

5. WBOS Control

Operational space control is used here for biped robots, which we model as a rigid multi-body system with a floating base and in contact with the environment.[15] A foot placement planner is proposed and stepping on unknown obstacle and push recovery are demonstrated. The WBOS framework is characterized by

$$A\ddot{q} + N_c^{\mathrm{T}}(b+g) + J_c^{\mathrm{T}}\Lambda_c\dot{J}_c\dot{q} = (SN_c)^{\mathrm{T}}\tau \tag{40}$$

where $q \in \mathbb{R}^{(n+6)}$ are generalized coordinates of the biped robot. n is the number of actuated degrees-of-freedom (DOFs). A is the mass inertia matrix, b is the centrifugal and Coriolis vector and, g is the gravity vector. J_c is the contact jacobian (left or right foot) and Λ_c is the kinetic energy matrix of the contact space. The generalized force in operational space is given by

$$F_{\text{task}} = \Lambda^*_{\text{task}} u_{\text{task}} + \mu^*_{\text{task}} + p^*_{\text{task}} \tag{41}$$

where F_{task} is the generalized force in task space. Λ^*_{task} is the kinetic energy matrix in task space. u_{task}, μ^*_{task}, and p^*_{task} are the task space acceleration, centrifugal and Coriolis force, and gravity force, respectively. u_{task}, μ^*_{task}, and p^*_{task} are given below:

$$\begin{cases} \Lambda^*_{\text{task}} = \left(J^*_{\text{task}} \Phi J^{*\mathrm{T}}_{\text{task}}\right)^{-1}, \\ \mu^*_{\text{task}} = \Lambda^*_{\text{task}} J^*_{\text{task}} \Phi b - \Lambda^*_{\text{task}} \dot{J}^*_{\text{task}} \dot{q}, \\ p^*_{\text{task}} = \Lambda^*_{\text{task}} J^*_{\text{task}} \Phi g \end{cases} \tag{42}$$

where $\Phi = (SN_c)A^{-1}(SN_c)^T$ is the constrained projection of A^{-1}. $J^*_{\text{task}} = J_{\text{task}}\overline{(SN_c)} \in \mathbb{R}^{n_{\text{task}} \times 6}$. For passive-foot biped robots with 6 DoFs, the task vector in the operational space is defined by the combined jacobian:

$$J_{\text{walking}} = \begin{bmatrix} J_{z_{\text{CoM}}} \\ J_{\text{posture}} \\ J_{\text{foot}} \end{bmatrix} \tag{43}$$

$J_{z_{\text{CoM}}}$, J_{posture}, and J_{foot} are the jacobians for the robot's CoM height, posture of the torso, and foot. The task acceleration is defined as:

$$u_{\text{walking}} = [\ddot{z}_{\text{CoM}}, \ddot{\theta}_{\text{pitch}}, \ddot{\theta}_{\text{roll}}, \ddot{x}_{\text{foot}}, \ddot{y}_{\text{foot}}, \ddot{z}_{\text{foot}}]^{\mathrm{T}} \tag{44}$$

where $\ddot{z}_{\text{CoM}}$, $\ddot{\theta}_{\text{pitch}}$, $\ddot{\theta}_{\text{roll}}$, $\ddot{x}_{\text{foot}}$, $\ddot{y}_{\text{foot}}$, $\ddot{z}_{\text{foot}}$ are the accelerations for CoM height, torso's pitch, and roll angles θ_{pitch}, θ_{roll}, foot position x_{foot}, y_{foot}, and z_{foot}.

In our task design, the CoM dynamics in the x and y directions are passive, which implies that it is intractable to control the desired motion of the x and y tasks and embed them in the whole body operational framework. In this study, a method is devised to plan a robust trajectory in the y and x phase space through adjusting the z trajectory. Here are our assumptions for the analysis of our CoM dynamic controller:

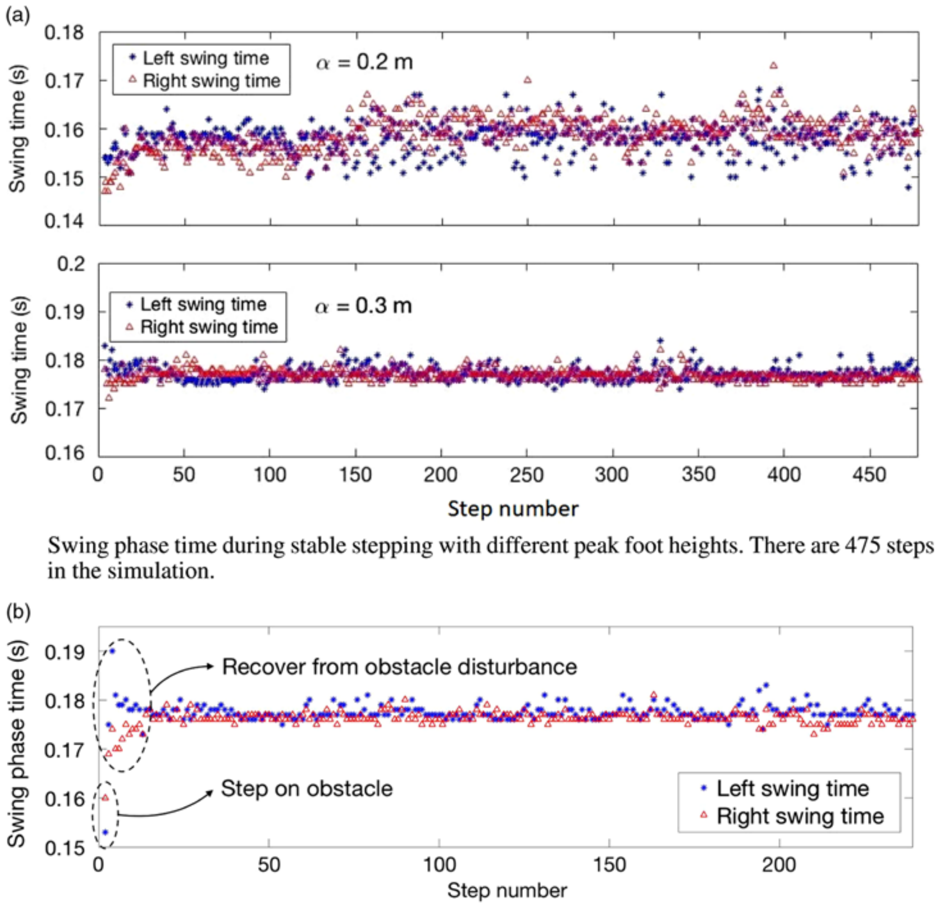
(1) The CoM angular momentum is ignored;
(2) There is no slip of the supporting foot.

The foot placement planner is designed to be:

$$\rho = \xi + \lambda \dot{\xi} \tag{45}$$

where $\rho \in \Omega = \{[x_{\text{foot}}, y_{\text{foot}}]^{\mathrm{T}} \in \mathbb{R}^2 | \quad ||r_{\text{foot}} - r_{\text{torso}}|| < l\}$, $r_{\text{foot}} = [x_{\text{foot}}, y_{\text{foot}}, z_{\text{foot}}]^{\mathrm{T}}$. r_{torso} is the origin of the torso's local frame. l is the maximum distance between the torso and the foot. The desired trajectory of the foot height is planned via a sine function. Figure 11(a) depicts the swing time in each phase when the biped robot stable step on a flat ground. The swing phase is switched based on the contact event rather than a fixed pre-planned phase time. Figure 11(b) shows an example when the robot steps on an unknown obstacle. The robot has no prior knowledge of the terrain. When the right foot lands on the unexpected obstacle, the left foot is considered for the update of the new foot placement. The simulation result shows that the robot displays robustness to the unknown terrains. The swing time is shortened when the foot lands ahead of time due to the obstacle. This can be seen in the first few points of Fig. 11(b).

Figure 12 demonstrates that the foot placement planner can recover from a push. The push comes from an external impact on the right side of the torso. The impact is 2100 N and lasts for 0.002 s, which is an impulse of 4.2 N · s. The impulse larger than 4.2 N · s generally leads to failure in push recovery due to kinematic constraints for maximum stepping length. The total mass of the simulated robot is around 30 kg, of which the torso, hip, thigh, and shank weigh is 15.1, 2.3, 3.5, and 1.1 kg, respectively. In Fig. 12, the red line depicts the evolution of the CoM state after the push. After a few foot placement adjustments, the COM state settles in a bounded region in the phase plane.



Swing phase time during stable stepping with different peak foot heights. There are 475 steps in the simulation.

Swing phase time when stepping on unexpected obstacle.There are 239 steps in the simulation. When the robot steps over an obstacle, the swing time is shortened because of the sudden landing.

Fig. 11. The evolution of swing phase time (a) during stable stepping and (b) during stepping on unexpected obstacle. The swing phase between left and right foots is switched based on contact events.

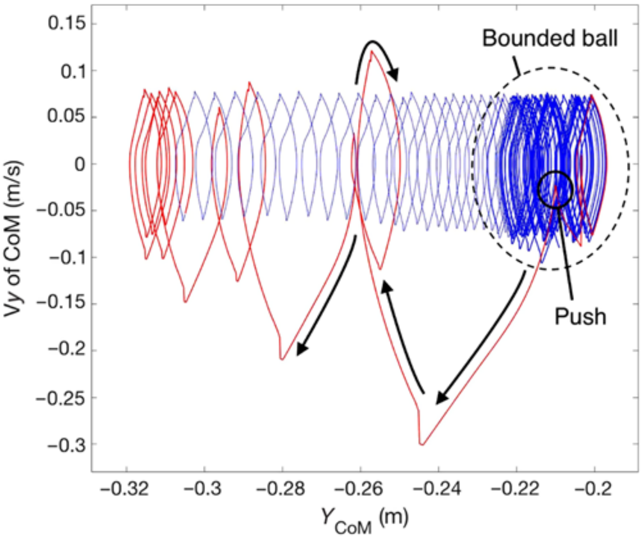


Fig. 12. The evolution of the position and velocity of the CoM in the *y* direction is shown here. The red line shows the evolution after an impulse is exerted on the robot. The CoM state moves in the direction of the black arrows. The blue lines are used to show when the CoM state converges back to a bounded ball.

6. Conclusion

In this study, a hierarchical control structure is considered to improve the robustness of bipedal locomotion. The hierarchical structure consists of three levels. At the high level, a midpoint-based method is proposed to generate a CoM trajectory that can track the path planned by $A*$. It is shown that the path planning method based on midpoints as feature is suitable for wave-shaped trajectories. At the middle level, a robust hybrid controller consisting of two sub-controllers is proposed. The simulation proves that the disturbed states in the extended phase space can be dragged back to the nominal trajectory planned by MPC without chattering issues. At the low level, the flexibility of a WBOS framework is put to use through designing CoM motion, and posture and foot placement trajectories as tasks. It is demonstrated in simulation that the motion controller displays robustness to unexpected obstacles and push disturbances. The three levels are independent with each other: the high level generates a feasible CoM trajectory and the middle-level controller provides robustness to the disturbances. The low level incorporates CoM motion as a task in the WBOS framework which increases the adaptability to uneven terrains.

Overall, this study demonstrates the effectiveness of the proposed hierarchical control structure through simulation. We are currently developing an experimental robot platform so that it can support large enough joint torque control bandwidth, accurate state estimation, and powerful actuation.[38] In the meantime, this study focuses on the theoretical developments and simulation validation. The simulation does not consider joint actuation latencies, thus assuming pure torque source in the WBOS framework. And the state and sensor noise are also ignored. State estimation is assumed to be accurate. In the real robot, we will encounter all of these additional difficulties

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant 61533004, U1613206, in part by Guangdong Innovative and Entrepreneurial Research Team Program under Grant 2016ZT06G587, in part by the Office of Naval Research, ONR grant #N000141512507, and in part by the NSF grant #1724360. The authors would like to thank Oussama Khatib, Jee-Hwan Ryu, Shameek Ganguly, Xiyang Yeh, Junkai Lu, Brian Soe, Samir Menon, and Hongkai Dai for their valuable input.

References

1. H. Dai, A. Valenzuela and R. Tedrake, "Whole-Body Motion Planning with Centroidal Dynamics and Full Kinematics," *IEEE-RAS International Conference on Humanoid Robots*, Madrid, Spain (2014) pp. 295–302.
2. S. Feng, E. Whitman, X. Xinjilefu and C. G. Atkeson, "Optimization Based Full Body Control for the Atlas Robot," *Proceedings of the 14th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, Madrid, Spain (2014) pp. 120–127.
3. J. Garimort, A. Hornung and M. Bennewitz, "Humanoid Navigation with Dynamic Footstep Plans," *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Shanghai, China (2011) pp. 3982–3987.
4. P. Michel, J. Chestnutt, J. Kuffner and T. Kanade, "Vision-Guided Humanoid Footstep Planning for Dynamic Environments," *Proceedings of the 5th IEEE-RAS International Conference on Humanoid Robots*, Seoul, Korea (2005) pp. 13–18.
5. L. Baudouin, N. Perrin, T. Moulard, F. Lamiraux, O. Stasse and E. Yoshida, "Real-Time Replanning Using 3d Environment for Humanoid Robot," *Proceedings of the 11th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, Bled, Slovenia (2011) pp. 584–589.
6. J. Chestnutt, K. Nishiwaki, J. Kuffner and S. Kagami, "An Adaptive Action Model for Legged Navigation Planning," *Proceedings of the 7th IEEE-RAS International Conference on Humanoid Robots*, Pittsburgh, USA (2007) pp. 196–202.
7. J. J. Kuffner, S. Kagami, K. Nishiwaki, M. Inaba and H. Inoue, "Dynamically-stable motion planning for humanoid robots," *Auton. Robots* **12**(1), 105–118 (2002).
8. J. Kuffner, K. Nishiwaki, S. Kagami, M. Inaba and H. Inoue "Motion Planning for Humanoid Robots under Obstacle and Dynamic Balance Constraints," *Proceedings of the IEEE International Conference on Robotics and Automation*, Seoul, Korea, vol. 1 (2001) pp. 692–698.
9. J. Chestnutt, M. Lau, G. Cheung, J. Kuffner, J. Hodgins and T. Kanade, "Footstep Planning for the Honda Asimo Humanoid," *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, ICRA, Barcelona, Spain (2005) pp. 629–634.

10. Y. Zhao, B. R. Fernandez and L. Sentis, "Robust optimal planning and control of non-periodic bipedal locomotion with a centroidal momentum model," *Int. J. Rob. Res.* **36**(11), 1211–1242 (2017).
11. E. R. Westervelt, C. Chevallereau, J. H. Choi, B. Morris and J. W. Grizzle. *Feedback Control of Dynamic Bipedal Robot Locomotion* (CRC Press, Boca Raton, USA, 2007).
12. M. Posa, M. Tobenkin and R. Tedrake, "Lyapunov Analysis of Rigid Body Systems with Impacts and Friction via Sums-of-Squares," *Proceedings of the 16th International Conference on Hybrid Systems: Computation and Control*, Philadelphia, USA (2013) pp. 63–72.
13. I. R. Manchester, M. M. Tobenkin, M. Levashov and R. Tedrake, "Regions of Attraction for Hybrid Limit Cycles of Walking Robots," *IFAC Proceedings Volumes*, **44**(1) (2011), pp. 5801–5806.
14. A. Papachristodoulou and S. Prajna, "Robust stability analysis of nonlinear hybrid systems," *IEEE Trans. Automat. Contr.* **54**(5), 1035–1041 (2009).
15. L. Sentis, Synthesis and Control of Whole-Body Behaviors in Humanoid Systems *Ph.D. Dissertation* (Stanford University, Stanford, 2007).
16. T. Koolen, S. Bertrand, G. Thomas, T. De Boer, T. Wu, J. Smith, J. Englsberger and J. Pratt, "Design of a momentum-based control framework and application to the humanoid robot atlas," *Int. J. HR* **13**(1), 1650007 (2016).
17. J. Pratt, C. M. Chew, A. Torres, P. Dilworth and G. Pratt, "Virtual model control: An intuitive approach for bipedal locomotion," *Int. J. Rob. Res.* **20**(2), 129–143 (2001).
18. O. Khatib and J. Burdick, "Motion and Force Control of Robot Manipulators," *Proceedings of the 1986 IEEE International Conference on Robotics and Automation*, San Francisco, USA, vol. 3 (1986) pp. 1381–1386.
19. J. Luo, Y. Zhao, D. Kim, O. Khatib and L. Sentis, "Locomotion Control of Three Dimensional Passive-Foot Biped Robot Based on Whole Body Operational Space Framework," *Proceedings of the 2017 IEEE International Conference on Robotics and Biomimetics (ROBIO)*, Marco, China (2017) pp. 1577–1582.
20. B. J. Stephens and C. G. Atkeson, "Dynamic Balance Force Control for Compliant Humanoid Robots," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Taipei, Taiwan (2010) pp. 1248–1255.
21. M. Posa, S. Kuindersma and R. Tedrake, "Optimization and Stabilization of Trajectories for Constrained Dynamical Systems," *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, Stockholm, Sweden (2016) pp. 1366–1373.
22. L. Righetti and S. Schaal, "Quadratic Programming for Inverse Dynamics with Optimal Distribution of Contact Forces," *Proceedings of the 12th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, Osakam, Japan (2012) pp. 538–543.
23. M. Johnson, B. Shrewsbury, S. Bertrand, T. Wu, D. Duran, M. Floyd, P. Abeles, D. Stephen, N. Mertins, A. Lesman and J. Carff, "Team IHMC's lessons learned from the DARPA robotics challenge trials," *J. Field Robot.* **32**(2), 192–208 (2015).
24. C. Ott, R. Mukherjee and Y. Nakamura, "A hybrid system framework for unified impedance and admittance control," *J. Intell. Robot. Syst.* **78**(3–4), 359–375 (2015).
25. S. Feng, E. Whitman, X. Xinjilefu and C. G. Atkeson, "Optimization-based full body control for the DARPA robotics challeange," *J. Field Robot.* **32**(2), 293–312 (2015).
26. S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi and H. Hirukawa, "The 3D Linear Inverted Pendulum Mode: A Simple Modeling for a Biped Walking Pattern Generation," *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, Maui, USA, vol. 1 (2001) pp. 239–246.
27. W. Gao, Z. Jia and C. Fu. "Increase the feasible step region of biped robots through active vertical flexion and extension motions," *Robotica* **35**(7), 1541–1561 (2017).
28. J. Pratt, T. Koolen, T. De Boer, J. Rebula, S. Cotton, J. Carff, M. Johnson and P. Neuhaus, "Capturability-based analysis and control of legged locomotion, Part 2: Application to M2V2, a lower-body humanoid," *Int. J. Rob. Res.* **31**(10), 1117–1133 (2012).
29. J. W. Luo, Y. L. Fu and S. G. Wang. "3D stable biped walking control and implementation on real robot," *Adv. Robot.* **31**(12), 634–649 (2017).
30. M. H. Raibert, *Legged Robots that Balance* (MIT Press, Cambridge, MA, 1986).
31. H. Geyer, A. Seyfarth and R. Blickhan, "Compliant Leg Behaviour Explains Basic Dynamics of Walking and Running," *Proceedings of the Royal Society B: Biological Sciences* (2006) pp. 2861–2867.
32. S. Rezazadeh, C. Hubicki, M. Jones, A. Peekema, J. Van Why, A. Abate and J. Hurst, "Spring-mass Walking with ATRIAS in 3D: Robust Gait Control Spanning Zero to 4.3 kph on a Heavily Underactuated Bipedal Robot," *ASME 2015 Dynamic Systems and Control Conference*, Columbus, USA (2015) pp. V001T04A003–V001T04A003.
33. H. R. Vejdani, Y. Blum, M. A. Daley and J. W. Hurst, "Bio-inspired swing leg control for spring-mass robots running on ground with unexpected height disturbance," *Bioinspir. Biomim.* **8**(4), 046006 (2013).
34. 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).
35. D. Kim, G. Thomas and L. Sentis, "Continuous Cyclic Stepping on 3D Point-foot Biped Robots via Constant Time to velocity Reversal," *Proceedings of the 13th International Conference on Control Automation Robotics & Vision (ICARCV)*, Singapore (2014) pp. 1637–1643.

36. P. B. Wieber, "Trajectory Free Linear Model Predictive Control for Stable Walking in the Presence of Strong Perturbations," *IEEE-RAS International Conference on Humanoid Robots*, Genoa, Italy (2006).
37. S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi and H. Hirukawa, "Biped walking pattern generation by using preview control of zero-moment point," *IEEE Trans. Ind. Electron.* **60**(11), 5137–5147 (2013).
38. J. Luo, S. Wang, Y. Zhao and Y. Fu, "Variable stiffness control of series elastic actuated biped locomotion," *Intel. Serv. Robot*. **11**(3), 225–235 (2018).