Robotica (2018) volume 36, pp. 57–77. © Cambridge University Press 2017
doi:10.1017/S0263574717000157

Footstep adaptation strategy for reactive omnidirectional walking in humanoid robots

Jiwen Zhang†‡§, Zeyang Xia¶, Li Liu†‡§ and Ken Chen†‡§*

†*Department of Mechanical Engineering, Tsinghua University, Beijing 100084, China.*
E-mails: jwzhang@mail.tsinghua.edu.cn, liuli@tsinghua.edu.cn, kenchen@tsinghua.edu.cn
‡*Beijing Key Lab of Precision/Ultra-precision Manufacturing Equipments and Control, Tsinghua University, Beijing 100084, China*
§*The State Key Laboratory of Tribology, Tsinghua University, Beijing 100084, China*
¶*Shenzhen Institutes of Advanced Technology, Chinese Academy of Sciences, Shenzhen 518055, China. E-mail: zeyang.xia@ieee.org*

(Accepted March 8, 2017. First published online: April 12, 2017)

SUMMARY
Stability, high response quality and rapidity are required for reactive omnidirectional walking in humanoids. Early schemes focused on generating gaits for predefined footstep locations and suffered from the risk of falling over because they lacked the ability to suitably adapt foot placement. Later methods combining stride adaptation and center of mass (COM) trajectory modification experienced difficulties related to increasing computing loads and an unwanted bias from the desired commands. In this paper, a hierarchical planning framework is proposed in which the footstep adaption task is separated from that of COM trajectory generation. A novel omnidirectional vehicle model and the inequalities deduced therefrom are adopted to describe the inter-pace connection relationship. A constrained nonlinear optimization problem is formulated and solved based on these inequalities to generate the optimal strides. A black-box optimization problem is then constructed and solved to determine the model constants using a surrogate-model-based approach. A simulation-based verification of the method and its implementation on a physical robot with a strictly limited computing capacity are reported. The proposed method is found to offer improved response quality while maintaining rapidity and stability, to reduce the online computing load required for reactive walking and to eliminate unnecessary bias from walking intentions.

KEYWORDS: Humanoid robots; Omnidirectional; Reactive walking; Inverted pendulum model; Nonlinear optimization; Surrogate model optimization.

Nomenclature

3D-LIPM	3D Linear Inverted Pendulum Model
COM	Center of mass
MPC	Model predictive control
IPCR	Inter-pace connection relationship
ZMP	Zero moment point
DACE	Design and analysis of computer experiment

1. Introduction

Omnidirectional walking means that a humanoid robot can take a step in any direction with any stride at will, thus allowing the robot to walk forward/backward, take a step to the side, turn to change

* Corresponding author. Email: kenchen@tsinghua.edu.cn

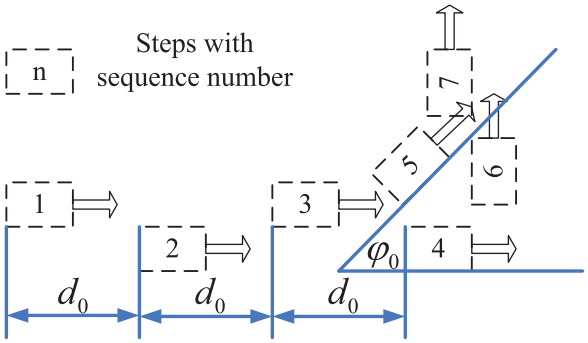


Fig. 1. An example of a step sequence issued to a joint trajectory generator, with a sharp turn representing a large stride variation.

direction, or perform any combination of these movements with any magnitude. This type of motion is important because the environments in which human beings live and work are filled with various movable objects and unstructured obstacles, and a robot operating in such an environment must be able to reactively avoid collisions with these obstacles.

There are several key factors that must be considered in a reactive omnidirectional walking engine. The first and most basic one is stability, which is strictly required for any type of motion planning for a humanoid robot. Some safety margin is necessary, which should be as large as possible to prevent the robot from falling over easily. The second factor to be considered is the response quality. The walking engine should be able to accept an arbitrary sequence of stride commands, either from a joystick, as in,[1] or from some other artificial intelligence module, such as a decision-making and path planning module,[2] and should be capable of generating joint trajectories in real time to follow these commands. The delay of the trajectory generator and the deviance of the generated motions from the expected motions must be kept as low as possible. The third factor of concern is rapidity. A slow walking style does not represent any significant improvement with regard to obstacle avoidance and navigation, and it reduces the necessity for omnidirectional reactive walking. All these considerations give rise to a need for additional focus on this specific problem.

Gait planning methods have reached an advanced stage of development over the years, and some of them can generate joint trajectories that can guarantee stability in real time. The 3D-LIPM (3D Linear Inverted Pendulum Model)[3] and the preview controller[4,5] are the most widely used methods. Using these methods, joint trajectories can be generated for any predefined foot placement sequence with very little computation, thereby providing a foundation for realizing higher levels of intelligence in walking behavior, such as obstacle avoidance and navigation.[6,7] However, the question of whether each such sequence can be successfully transformed into a stable and rapid walking gait requires further investigation. Suppose that the footsteps depicted in Fig. 1, which include a sharp turn of 90° connecting two sections of straight walking, are issued as the walking intentions. A robot can easily follow these footsteps at a slow walking speed, e.g., at 2 s/step, but may suffer from a high risk of falling over when it is forced to complete each stride at 0.2 s/step because, in this case, the robot has insufficient time to modify its walking velocity to turn 90° due to inertial effects. Although human beings are able to slow down specific individual steps to negotiate large stride changes, it appears difficult for a present-day gait planner to achieve the same goal because complex dynamics analysis is required,[8] which is unfeasible for real-time computing.

Herdt *et al.*[9] claimed that a means of stride adaptation is needed to eliminate the effect of large velocity variations and replaced the desired footstep sequence provided as the input to the gait generator with a reference speed vector, thereby allowing the robot to track such input speed vectors as closely as possible; however, some bias from the issued commands may remain if they are not reasonable with regard to walking stability. However, the model predictive control (MPC) method presented in ref. [9] has certain other shortcomings: the computational burden is much heavier,[10] and unnecessary bias from the intended velocity vectors exists,[11] which may decrease the response quality and rapidity of reactive omnidirectional walking.

This paper seeks to provide another solution specifically to the task of tracking walking intentions. A hierarchical planning framework and an omnidirectional moving vehicle model are used. This

new model first appeared in ref. [12]; however, the proposal given at that time resorted to invoking empirical experience without making full use of the generated inequalities, and no well-defined theoretical solution was provided. Moreover, the hidden model parameters also were not determined. In this paper, these problems are addressed via two rounds of optimization: the first is a small-scale nonlinear optimization problem, which is solved online within the time cycle of a single step, and the second is a surrogate-model-based black-box optimization problem to determine the most suitable constants for the dynamics model and is solved offline.

The remainder of this paper is organized as follows. Section 2 reviews the state-of-the-art results related to this research. Section 3 outlines the proposed hierarchical planning framework and introduces the omnidirectional vehicle model and the obtained constraint inequalities, termed the inter-pace connection relationship (IPCR); then, in Section 4, the task of footstep determination is investigated in detail, whereas Section 5 describes the derivation of the model constants via black-box optimization. Section 6 presents a comparison of footstep adaptation strategies and analyzes the overall applicability of the proposed strategy. Section 7 reports the application of the proposed method in simulations and physical robot experiments, and conclusions are presented in Section 8.

2. State of the Art

The generation of walking gaits for humanoids has been a focus of research for several decades, and there is already a rich body of literature regarding basic walking skills. In one subset of these studies, the robot is regarded as a multi-rigid-body dynamic system, as in refs. [13–15]; however, the heavy computing loads incurred by the optimization and search processes used in these methods prevent their use in reactive walking.

Another set of studies use simplified dynamics models, such as the 3D-LIPM or the cart-table model proposed by Kajita *et al.*[3,4] In this model, the robot's mass is concentrated at its center of mass (COM), which is connected to a massless rod representing its leg. The rod is extensible to allow the COM to move within a plane. Because of the linear dynamics function, the COM trajectory and, therefore, the joint trajectories can be obtained with much less effort than in the multi-rigid-body approach. A preview controller based on the 3D-LIPM has been demonstrated in studies by Kajita *et al.*,[4,5] and direct analytical solutions to the differential equations of the 3D-LIPM are provided in ref. [16]. Later, omnidirectional reactive walking was achieved using both approaches, as in refs. [17–22].

Wieber[23] has noted that Kajita's method does not explicitly consider the satisfaction of the zero moment point (ZMP) criterion,[24] which allows walking stability to be determined by calculating the point of action of the contact forces exerted by the ground. Instead, Kajita's method merely minimizes the deviation between the prescribed and planned ZMP trajectories. The robot tends to fall over when a large variation in the COM velocity occurs, which can be caused by the action of an external force as well as by abrupt changes in walking intentions, as in Fig. 1. Wieber[23] therefore proposed the MPC method, a constrained quadratic optimization approach with explicit ZMP constraints for increased robustness. This method can also perform more advanced online COM trajectory modification because it can handle uncertain errors and disturbances through dynamic feedback.

Later, in Herdt's research,[9] the elimination of larger velocity variations was achieved using a modified version of the MPC method in which the deviation between the foot placement and the reference speed vector is minimized by adding a term representing it, weighted by a coefficient, to the objective function, which initially contained only the COM state variable, as formulated by Wieber.[23] Herdt[25] also extended the method by including turning steps.

The MPC scheme appears quite versatile, and it has even been used in visual servoing by incorporating percept information directly into the optimization objective, as done by Garcia *et al.*;[26] however, its computational burden is heavier than that of the previous strategy. Dimitrov *et al.*[27] developed an efficient solver specifically for the MPC optimization problem that offers valuable improvements. Moreover, Dimitrov *et al.*[11] found that coupling the tasks of generating the foot landing positions and the COM trajectories may always introduce an unnecessary bias from the intended velocity vectors. Therefore, he proposed the modified $l1$-norm and $l\infty$-norm objective functions, which reduce this effect to some extent but seem unable to eliminate it.

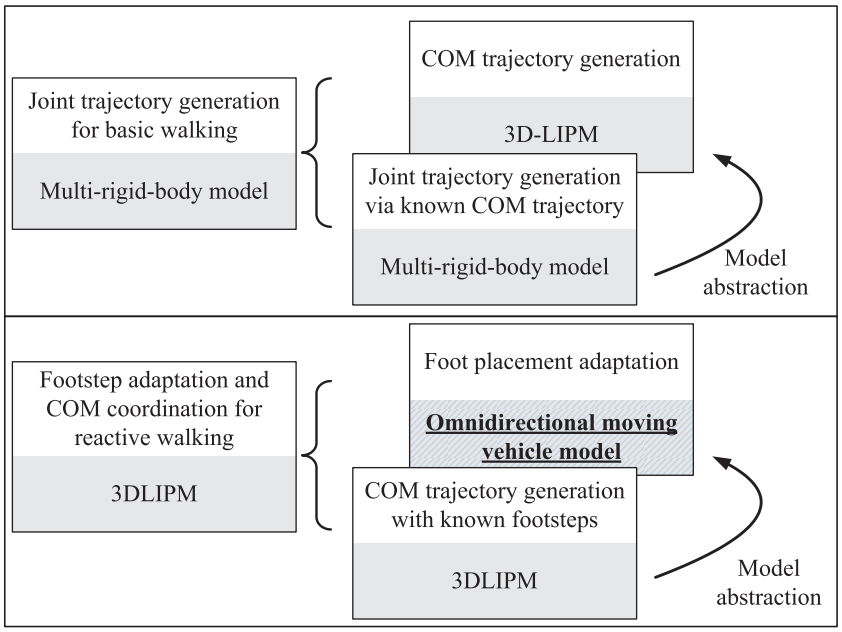


Fig. 2. Comparison of two model abstractions and the corresponding hierarchical planning schemes: the decoupling of the generation of the COM trajectories and joint trajectories via the 3D-LIPM (upper part) and the decoupling of the foot placement adaptation and the COM trajectory generation via the omnidirectional vehicle model (lower part).

3. The Omnidirectional Vehicle Model and the Inter-Pace Connection Relationship

3.1. The hierarchical planning strategy

The two key factors for joint trajectory generation are the footstep locations and the COM trajectories. The preview controller based on the 3D-LIPM that has previously been proposed by Kajita[4] requires predefined footsteps for the planning of COM trajectories, but rational predefined footsteps cannot be obtained if a stride adaptation problem arises as a result of the dynamics of the generated COM trajectories; consequently, these two factors are coupled. This problem is severe when the walking speed is high and the walking intentions are arbitrary. The entire problem can be directly solved using the MPC method by combining both factors into a single optimization objective, or, as an alternative approach, the problem can be hierarchically split to address these two factors separately, as in ref. [26].

A new model called the omnidirectional vehicle model provides guidance for building such a hierarchical framework. This concept originates from an abstraction and simplification of a humanoid robot. The 3D-LIPM is used to relieve the heavy computational burden of the multi-rigid-body approach, such as in Huang *et al.*,[13] because this model allows the COM trajectory to be obtained directly before any joint movements are considered. This is helpful for splitting the coupled problem of trajectory planning into a layered framework, as illustrated in the upper part of Fig. 2, although the dynamics of the 3D-LIPM differs considerably from that of a real robot.

Following a similar approach as before, we then attempt to further simplify the 3D-LIPM to create a new model to relieve the heavy computing burden of achieving footstep adaptation and COM coordination in a coupled approach via the 3D-LIPM. In this new model, the upper layer is dedicated to separately adapting the footstep locations, whereas the 3D-LIPM and the preview controller remain unchanged in the bottom layer. The process and the corresponding planning framework are illustrated in the lower part of Fig. 2.

3.2. The omnidirectional vehicle model

The omnidirectional vehicle model (see Fig. 3) has following features: A ball of radius r and evenly distributed mass m is fixed on a massless rod with a height of h from the flat ground, and this rod

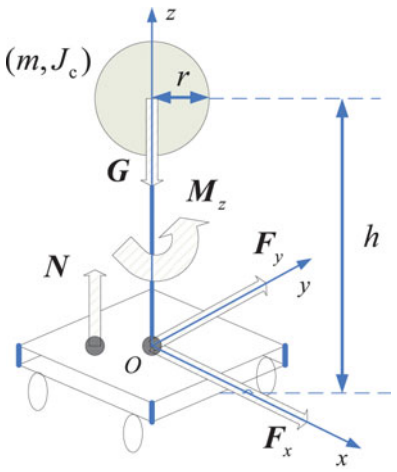


Fig. 3. Omnidirectional vehicle model and the corresponding force analysis.[12]

is also mounted on a massless omnidirectional moving vehicle at its other end; the overall moment of inertia is J_c, which originates from the mounted ball. Suppose that the vehicle is equipped with an internal engine and can move by itself, just like a car, without any external driving system. The vehicle must not tip over; this represents the stable motion of the corresponding humanoid robot.

Several considerations arise with this simplification. First, side-swinging movements and oscillations in the velocity of the robot's body tend to decrease when the walking velocity and step frequency increase. This phenomenon can be observed in human beings. A comparison of different step periods using the 3D-LIPM to generate COM trajectories also reveals a similar trend. As a result, when a humanoid robot is walking rapidly, its body movements can be approximated as steady rather than periodically oscillating. The body of a legged robot then appears similar to that of a moving vehicle, and its legs can be regarded as just another specialized driving system, such as wheels.

Second, this model is essentially an inverted pendulum model, and models of this type have previously been proven to be effective in motion planning and stabilization for bipedal robots. The vehicle model is unique because its base is constantly moving: accelerating, decelerating, and turning at will, which can be used to model continuous walking. By contrast, in other inverted pendulum models, the base is always assumed to be stationary. These models, therefore, are good at modeling a single walking step, but several such steps must then be consecutively connected to analyze continuous walking, which adds complexity.

Third, the dynamics of the concentrated mass at the COM plays a key role in defining the footsteps. Many previous studies have relied on such considerations, including studies on push recovery[28] and footstep adaption using the MPC method. The final outcome can be regarded as a set of carefully planned footsteps that track the desired COM motions while satisfying stability constraints. In the omnidirectional moving vehicle model, the dynamics of this concentrated mass is simulated by the mounted ball, whereas the stability constraints are approximated by the shape of the supporting vehicle. If we place such a vehicle into the scenario specified by Fig. 1, it must decelerate before taking the specified sharp turn to avoid tipping over. This process can be translated into the adaption of the footsteps by decreasing the stride, as expected.

The set of dynamics functions, the set of force synthesis functions and the geometric relationships that exist in the model are expressed by Eqs. (1)–(3), respectively:

$$\begin{aligned} a_x &= \dot{v}_x + \omega \cdot v_y \,, \\ a_y &= \dot{v}_y + \omega \cdot v_x \,, \\ \varepsilon &= \dot{\omega} \,; \end{aligned} \tag{1}$$

$$\begin{aligned} F_x &= m \cdot a_x \,, \\ F_y &= m \cdot a_y \,, \\ M_z &= J_{\mathrm{C}} \cdot \varepsilon \,; \end{aligned} \tag{2}$$

$$\begin{aligned} F_z &= N + G \,, \\ M_x &= N \cdot y_N + F_y \cdot h \,, \\ M_y &= N \cdot x_N + F_x \cdot h \,, \end{aligned} \tag{3}$$

where (v_x, v_y) represents the translational velocity; ω is the angular velocity; ε is the derivative of the angular velocity; (a_x, a_y) represents the acceleration of the COM in the forward and lateral directions, described in the inertial reference system O–xyz; F_x, F_y, and M_z together denote the equivalent net force acting on the model in the horizontal direction, whereas F_z, M_x, and M_y denote the equivalent net force in the vertical direction; G is the gravity acting on the vehicle; and N denotes the reaction force from the supporting ground, which is considered to act on the point (x_N, y_N), where the net moments $M_x = 0$ and $M_y = 0$ hold. Because there is no translational motion in the vertical direction, $F_z = 0$ also holds. When (x_N, y_N) is located within the supporting area of the vehicle, the necessary reaction force N can be supplied and the model moves stably; otherwise, the model begins to tip over because the contact with the ground can provide only a pushing force.

If the vehicle is assumed to be rectangular in shape for simplicity of calculation, then the constraints will be expressed as in Eq. (4):

$$\begin{aligned} x_N^{\min} &< x_N < x_N^{\max} \,, \\ y_N^{\min} &< y_N < y_N^{\max} \,. \end{aligned} \tag{4}$$

The evolution of the above set of functions will yield constraints on the translational and rotational movements of the system when J_{c} is ignored; however, the rotational acceleration must also be limited by the finite force that the ground can supply to increase the self-spin. Therefore, μ is defined as the ratio of the spin moment to the gravitational force, as shown in Eq. (5), and μ should lie below a certain threshold, as also indicated in Eq. (5), to avoid slippage on the ground:

$$\begin{aligned} \mu &= \left\| \vec{M}_z \right\| \Big/ \left\| \vec{G} \right\| \,, \\ \mu &< \mu_{\max} \,. \end{aligned} \tag{5}$$

Although the COM trajectory of a robot cannot have a constant acceleration during a step, the vehicle is assumed to move with a piecewise constant acceleration and angular acceleration for mathematical simplicity. This is an acceptable simplification because we are interested in the total translation distance and turning angle within the time duration of a walking step (i.e., a stride) rather than the detailed motion trajectory of the robot.

Analytical integration and approximation of the above set of functions, constraints, and assumptions ultimately yield inequalities related to the IPCR, as shown in Eq. (6); further details of the derivation are presented in Appendix A:

$$\begin{aligned} g x_N^{\min} (t_{k+1} - t_k)^2 \big/ h &< -(2\Delta\chi + \varphi\gamma) < g x_N^{\max} (t_{k+1} - t_k)^2 \big/ h \,, \\ g y_N^{\min} (t_{k+1} - t_k)^2 \big/ h &< -(2\Delta\gamma + \varphi\chi) < g y_N^{\max} (t_{k+1} - t_k)^2 \big/ h \,, \\ \| \Delta\varphi \| &< \frac{5g}{4r^2} \mu_{\max} (t_{k+1} - t_k)^2 \,, \end{aligned} \tag{6}$$

where $(t_{k+1} - t_k)$ is the time duration of one step, h is the height of the COM, g is the gravitational constant, (χ, γ) represents the forward and lateral components of the translational stride, φ is

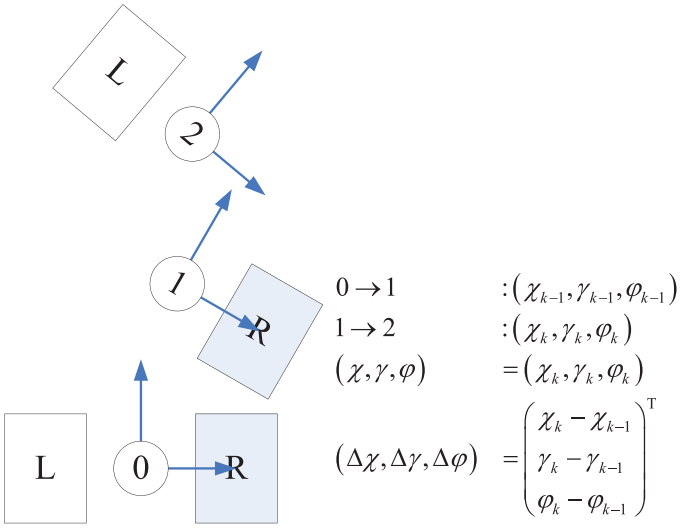


Fig. 4. Strides and their variations between consecutive steps.

the bearing angle of a turning stride, and $(\Delta\chi, \Delta\gamma, \Delta\varphi)$ represent the corresponding variations between consecutive steps. A stride is essentially a transformation of parameters between two sets of coordinates, defined with reference to an origin point at an offset from the landing foot of half the distance between the two feet when the robot is standing still. Figure 4 illustrates the definition of strides and the variation using two consecutive footsteps as examples.

When consecutive strides satisfy the IPCR defined in Eq. (6), they are considered to be reasonable; otherwise, they may cause the robot to tip over.

4. Omnidirectional Foot Placement Adaptation

When a humanoid robot is walking omnidirectionally, a footstep cannot be placed in the desired location if the constraints of Eq. (6) are not met; instead, it must be altered with respect to the previous stride to achieve the smallest possible deviation from the intentions. All these considerations together form an optimization problem.

4.1. Derivation of the optimization problem

The objective function of the optimization problem is expressed as the weighted sum of the squares of the deviations between the planned and commanded strides (see Eq. (7)). A lower value of δ in Eq. (7) indicates that the robot is exhibiting less error in tracking the commands it has been issued:

$$\delta = \lambda_\chi(\chi_c - \chi_k)^2 + \lambda_\gamma(\gamma_c - \gamma_k)^2 + \lambda_\varphi(\varphi_c - \varphi_k)^2 , \tag{7}$$

where χ_c, γ_c, and φ_c represent the commanded stride; χ_k, γ_k, and φ_k represent the corresponding planned stride to be optimized; and λ_χ, λ_γ, and λ_φ are the weights specifying the relative preferences for the individual components, where a higher value of λ implies that less error is desired for that component. The λ values are fixed and determined beforehand to prioritize either translational or turning movement.

In addition to the IPCR described by Eq. (6), the new stride must also satisfy certain geometric and kinematic constraints. A combined elliptical boundary is assigned for the forward/backward and lateral components, and an absolute threshold is imposed on the orientational variation; these constraints are collectively expressed in Eq. (8):

$$\begin{aligned} &|\varphi_k| \leqslant \varphi_{\max} , \\ &\frac{\chi_k^2}{\chi_{\max}^2} + \frac{\gamma_k^2}{\gamma_{\max}^2} \leq 1 , \end{aligned} \tag{8}$$

where $\chi_{\max}$, $\gamma_{\max}$, and $\varphi_{\max}$ are the maximum possible values of χ, γ, and φ, respectively, and act as stride limits. The resulting nonlinear optimization problem is summarized in Eq. (9):

$$\begin{aligned}
\min \quad & \lambda_\chi(\chi_c - \chi_k)^2 + \lambda_\gamma(\gamma_c - \gamma_k)^2 + \lambda_\varphi(\varphi_c - \varphi_k)^2 \\
\text{s.t.} \quad & 2\chi_k - 2\chi_{k-1} + \varphi_k\gamma_k + 2k_1\chi_{\max} \le 0\,, \\
& -2\chi_k + 2\chi_{k-1} - \varphi_k\gamma_k - 2k_2\chi_{\max} \le 0\,, \\
& 2\gamma_k - 2\gamma_{k-1} + \varphi_k\chi_k + 2k_3\gamma_{\max} \le 0\,, \\
& -2\gamma_k + 2\gamma_{k-1} - \varphi_k\chi_k - 2k_3\gamma_{\max} \le 0\,, \\
& \frac{\chi_k^2}{\chi_{\max}^2} + \frac{\gamma_k^2}{\gamma_{\max}^2} - 1 \le 0\,, \\
& \max(\varphi_{k-1} - k_5\varphi_{\max}, -\varphi_{\max}) \le \varphi_k\,, \\
& \varphi_k \le \min(\varphi_{k-1} + k_5\varphi_{\max}, \varphi_{\max})\,, \\
& -\chi_{\max} \le \chi_k \le \chi_{\max}\,, \\
& -\gamma_{\max} \le \gamma_k \le \gamma_{\max}\,.
\end{aligned} \tag{9}$$

The first through fourth constraint functions arise from the first two inequalities in Eq. (6), where k_1, k_2, and k_3 are expected to be known prior to optimization:

$$\begin{aligned}
k_1 &= \frac{g x_N^{\min}(t_{k+1} - t_k)^2}{h} \cdot \frac{1}{2\chi_{\max}}\,, \\
k_2 &= \frac{g x_N^{\max}(t_{k+1} - t_k)^2}{h} \cdot \frac{1}{2\chi_{\max}}\,, \\
k_3 &= \frac{g y_N^{\min}(t_{k+1} - t_k)^2}{h} \cdot \frac{1}{2\gamma_{\max}}\,.
\end{aligned} \tag{10}$$

Notably, humanoids are always assumed to exhibit the same motion capabilities for sidestepping left and right but may not have the same capabilities for stepping forward and backward. For this reason, the first pair of constraint functions contains two parameters, k_1 and k_2, to represent different forward and backward motion capabilities, but the second pair contains only one, k_3, to represent a symmetric sidestepping capability (i.e., $g y_N^{\min} = -g y_N^{\max}$). This also reflects the shape of the vehicle in Fig. 3, which is symmetric about the x axis but asymmetric about the y axis.

The sixth and seventh constraints on φ_k originate from the third inequality of Eqs. (6) and (8); they are combined because both limit the range of the next turning step. The parameter k_5 that appears in these two constraints is described by the following expression:

$$k_5 = \frac{5g}{4r^2}\mu_{\max}(t_{k+1} - t_k)^2 \cdot \frac{1}{\varphi_{\max}}\,. \tag{11}$$

The eighth and ninth constraints arise naturally from the valid ranges of the fifth constraint function.

Thus, the next footstep that best conforms to the desired stride is obtained by solving a quadratic optimization problem with five nonlinear inequality constraints, as expressed in Eq. (9). The optimization problem considered here involves only three variables, and the gradients for both the objective function and the constraints can be easily obtained despite their nonlinearity. There are several techniques that can be used to solve this problem with high efficiency, such as penalty-function-based methods, Sequential Quadratic Programming (SQP) or the Method of Moving Asymptotes (MMA).[29] The Augmented Lagrange Method,[30] a penalty-function-based approach, was chosen for our simulations and experiments because of the availability of its source code.

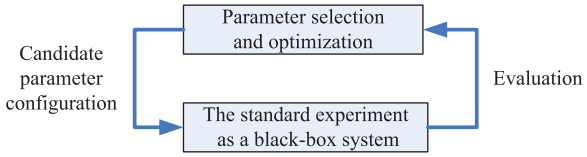


Fig. 5. Framework of the black-box optimization problem for determining the model constants.[32]

4.2. Discussion of the model constants

As is evident from Eq. (9), we obtain a new set of constants, namely, k_1, k_2, k_3, k_5, $\gamma_{\max}$, $\chi_{\max}$, and $\varphi_{\max}$, through transformation from the physical quantities that appear in the omnidirectional vehicle model. This mathematical procedure is done for the following reason: although the virtual height of the centralized mass can be simply set equal to the height of the COM, and the radius can be calculated from the inertia of the centralized mass around the COM as described by Orin,[31] the size of the vehicle is extremely difficult to determine. It has no direct physical correspondence with a robot; as mentioned before, it is only a representation of the robot's vulnerability to stride variations. By contrast, the new constants after transformation can be more easily determined from walking experiments, as discussed in Section 5.

Moreover, the dimensionless ratios k_1, k_2, k_3 and k_5 in Eq. (9) have specific physical meanings. For example, the first constraint function in Eq. (9) simplifies to Eq. (12) if the robot does not take any turning steps:

$$\chi_k - \chi_{k-1} \leqslant -k_1 \chi_{\max} \,. \tag{12}$$

When $-1 < k_1 < 0$, the robot is not able to reach its maximum stride, i.e., $\chi_{\max}$, within a single step and must instead accelerate over several steps. By contrast, when $k_1 < -1$, the robot is able to reach $\chi_{\max}$ within a single step. Similar conditions apply for k_2, k_3, and k_5. Therefore, these transformed constants are essentially equivalent to the ratios between the maximum stride and the translational acceleration limits.

These constants are essential for the omnidirectional vehicle model because they serve as the bounds that define valid consecutive steps. The robot will either lose its stability or adopt an over-conservative strategy during rapid reactive motion if the values of these constants are not chosen appropriately.

5. Determination of the Model Constants

An optimization problem must be solved to determine the best configuration to achieve the best reactive omnidirectional tracking quality within the robot's motion capabilities. However, this is a complex task for several reasons.

First, no well-defined criteria exist for evaluating the quality of reactive omnidirectional walking; such criteria must consider at least the response quality, stability and walking speed, as mentioned in Section 1, and should reflect the trade-off relationship among these three factors. Additionally, the relationships between the seven model constants and the robot's walking capability are also unavailable. These two facts imply that the objective function to be optimized cannot be determined.

Second, it is also difficult to obtain the constraint functions for the specific problem at hand. The relevant constraints must include at least the kinematic feasibility and dynamic stability. However, the constraint functions are not scalable before the completion of the entire task of omnidirectional walking, which is arbitrary and unpredictable.

Instead of struggling to express the problem explicitly, we can attempt to empirically derive these parameters by performing specific experiments, in which the robot is continuously run through the walking process using the same sets of velocity commands but different parameter configurations. By recording the results and comparing them with each other, the optimal parameters can ultimately be selected. The entire process can be summarized as a black-box optimization process, as illustrated in Fig. 5, which contains no mathematical expressions for either the objective or the constraints. Hemker *et al.*[32] have previously presented an example of black-box optimization for humanoid walking speed enhancement.

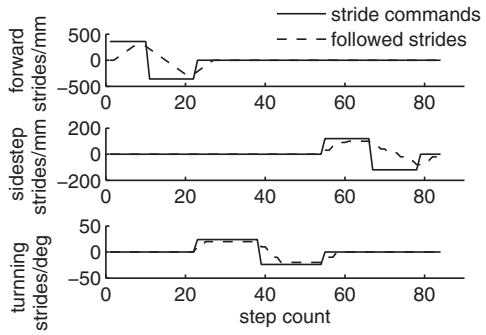


Fig. 6. Stride commands issued in a standard experiment and the results of following these commands.[12]

5.1. Evaluation of reactive walking

The tracking error incurred during a particular experiment is a potential measure of performance in reactive omnidirectional walking. The experimental process considered in Zhang *et al.*,[12] which comprises the series of desired stride commands depicted in Fig. 6, is adopted as the standard experiment; the robot is expected to walk at its maximum forward stride; then to directly shift to its maximum backward stride, its maximum leftward turning stride, and its maximum rightward turning stride, in sequence; and finally to sidestep to its left- and right-hand sides, one after the other. This standard process cannot replicate every possible omnidirectional condition, but it includes sharp transitions in all three stride components (i.e., χ, γ, and φ) and thus can indicate the response quality during such transitions. The tracking error with respect to the independent maximum stride commands χ, γ, and φ tends to drive a compromise between the response quality and the maximum walking speed. Finally, the evaluation is effective only if the entire walking experiment represented in Fig. 6 can be stably completed, which reflects the stability consideration. Therefore, this standard experiment accounts for all three basic requirements of omnidirectional reactive walking while remaining simple to perform using a robot.

The errors on all three components are squared, weighted and summed, as shown in Eq. (7), to serve as the objective function in Eq. (9), and these errors are then summed over every step and normalized, as shown in Eq. (13). A lower value of this measure indicates better tracking of the walking intentions:

$$\Delta = \frac{\sum \delta_k}{\sum \left(\lambda_\chi \chi_c^2 + \lambda_\gamma \gamma_c^2 + \lambda_\varphi \varphi_c^2\right)}, \tag{13}$$

where δ_k for each step during the standard experiment is expressed as in Eq. (7). The denominator of Eq. (13) is assumed to correspond to the condition in which there is no valid response stride for the walking generator (i.e., χ_k, γ_k, and φ_k are all zero regardless of the stride command) and acts as a normalization quantity to facilitate comparison among all experiments. If the standard experiment is successful, without the robot falling over before it reaches the end of the entire walking procedure, then Eq. (13) is used as the score. However, if the robot falls over at any point, then a value of $+\infty$ is recorded as the score. The objective is then to find the configuration of constants with the minimum score.

5.2. Black-box optimization using a surrogate-model-based approach

The seven effective constants discussed in Section 4.2, which can be collectively expressed as a vector $\boldsymbol{x}$, together with the experimental results presented in Eq. (13), establish a mapping. Given Ω as the valid range of $\boldsymbol{x}$, the black-box optimization problem can be described as follows:

$$\max_{\boldsymbol{x}\in\Omega} \Delta(\boldsymbol{x}),\ \Delta : \mathrm{R}^7 \to \mathrm{R}\,. \tag{14}$$

Candidate solvers for this problem can be broadly categorized into three different types: the stochastic-based approach, the sampling-based approach, and the surrogate-model-based approach.[32]

Because the exhaustive experiments required by the first two types of methods may destroy the robot before the optimal values can be obtained, the final approach appears to be the best candidate for solving this particular problem because it imposes far fewer demands in terms of experimental evaluation. The Design and Analysis of Computer Experiments (DACE) approach established by Sacks[33,34] is a widely used method of this type.

DACE relies on modeling a stochastic process called Kriging to approximate the behavior of an objective function with high nonlinearity. The basic function is expressed as follows:

$$y(\boldsymbol{x}) = \boldsymbol{f}(\boldsymbol{x})^{\mathrm{T}}\boldsymbol{\beta} + z(\boldsymbol{x})\,, \tag{15}$$

where $\boldsymbol{f}(\boldsymbol{x})$ is the selected base function and $z(\boldsymbol{x})$ is a variable corresponding to the stochastic process of interest, which follows a normal distribution with an expectation value of zero and a variance of σ. The values of the stochastic variable $z(\boldsymbol{x})$ at different positions $\boldsymbol{x}$ share the same distribution but are not independent; the covariance and correlation coefficient between any two points $\boldsymbol{x}^{(i)}$ and $\boldsymbol{x}^{(j)}$ are described as follows:

$$\begin{aligned}
&\operatorname{cov}\left[z\left(\boldsymbol{x}^{(i)}\right), z\left(\boldsymbol{x}^{(j)}\right)\right] = \sigma^2 R\left(\boldsymbol{\theta}, \boldsymbol{p}, \boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}\right)\,, \\
&r_i\left(\boldsymbol{x}^{(j)}\right) = \operatorname{corr}\left[z\left(\boldsymbol{x}^{(i)}\right), z\left(\boldsymbol{x}^{(j)}\right)\right] = R\left(\boldsymbol{\theta}, \boldsymbol{p}, \boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}\right)\,.
\end{aligned} \tag{16}$$

$R(\boldsymbol{\theta}, \boldsymbol{p}, \boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)})$ denotes the type of correlation and is defined as

$$R(\boldsymbol{\theta}, \boldsymbol{p}, \boldsymbol{x}^{(i)}, \boldsymbol{x}^{(j)}) = \exp\left(-\sum_{h=1}^{k}\theta_h\left|x_h^{(i)} - x_h^{(j)}\right|^{p_h}\right)\,, \tag{17}$$

where θ_h denotes the correlation strength of each x_h component and p_h can take any value on the interval [1, 2]. Equations (15)–(17) describe the DACE stochastic process model, or the Kriging model.[34] The parameters of this model, $\left\{\beta, \sigma^2, \boldsymbol{\theta}, \boldsymbol{p}\right\}$, are found via maximum likelihood estimation based on sufficient experimental points. Simultaneously, the mean square error (MSE) of the estimation can also be obtained using Eq. (18):

$$\begin{aligned}
&\hat{y}(\boldsymbol{x}^*) = \hat{\beta} + \boldsymbol{r}^{\mathrm{T}}\boldsymbol{R}^{-1}(\boldsymbol{y} - \boldsymbol{1}\cdot\hat{\beta})\,, \\
&s^2(\boldsymbol{x}^*) = \sigma^2\left[1 - \boldsymbol{r}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{r} + \frac{\left(1 - \boldsymbol{1}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{r}\right)^2}{\boldsymbol{1}^{\mathrm{T}}\boldsymbol{R}^{-1}\boldsymbol{1}}\right]\,,
\end{aligned} \tag{18}$$

where $\hat{\beta}$ is the estimated coefficient of the base function (which is always a constant value without the involvement of $\boldsymbol{x}$); $\boldsymbol{R}$ is the matrix of the correlation functions of the experimental points, with $r_i\left(\boldsymbol{x}^{(j)}\right)$ in Eq. (16) denoting the element in the ith row and jth column; $\boldsymbol{r}$ is the vector of the correlation functions of $\boldsymbol{x}^*$ and the experimental points; and $\boldsymbol{y}$ is the result vector for the experimental points.

This stochastic process model interpolates the experimental points using a continuously derivable function, whose minimum value can be determined numerically using a nonlinear planning algorithm. In other words, the mapping of the seven effective parameters to the evaluation result, for which an explicit mathematical expression was initially lacking, is replaced by a surrogate function, allowing the problem to be solved using classical nonlinear optimization methods.

In addition to the basic Kriging modeling process used in DACE, globally and locally balanced sampling strategies are realized using the expected improvement (EI) function,[34] and an experimental design generated using the Latin hypercube approach[35] is also adopted for the initial surrogate model. The entire iterative algorithm is summarized in Algorithm 1.

Algorithm 1 Iterative DACE optimization for model constants

Create N initial points $\boldsymbol{x}^{(i)}$ using Latin hypercube sampling;
Evaluate $\Delta(\boldsymbol{x}^{(i)})$ in Eq. (13) by means of walking experiments, $i = 1, 2 \ldots N$;
while iteration not terminated **do**
    Build surrogate model by running DACE using the set of points $\boldsymbol{x}^{(i)}$, $i = 1, 2 \ldots N$;
    Search for the maximum $\hat{\boldsymbol{x}}^{\star}$ of the EI function proposed by Jones et al.;[34]
    Evaluate $\Delta(\hat{\boldsymbol{x}}^{\star})$ in Eq. (13) by means of walking experiments;
    Add $\hat{\boldsymbol{x}}^{\star}$ to the point set;
    $N = N + 1$;
    Search for the minimum $\hat{\boldsymbol{x}}^{\star\star}$ of $\Delta(\boldsymbol{x})$ in Eq. (13) and output as the current result;
end while

This type of surrogate-model-based optimization is performed offline; thus, although it is time-consuming, it is not computationally demanding during reactive walking.

6. Comparison and Applicability of the Strategy

Before the simulations and experiments are presented, a brief comparison of different footstep adaption strategies is offered and the applicability of the overall proposal is discussed.

6.1. Comparison with the MPC approach

The two objectives of the hierarchical framework are to relieve the computational burden and eliminate the unwanted bias from the issued walking intentions that arise in the coupled approach.

The latter objective can be achieved by means of the new planning framework: when the commanded stride remains unchanged and the robot's walking stride has also reached an unchanging state, the IPCR simplifies to

$$
\begin{aligned}
& g x_N^{\min}(t_{k+1} - t_k)^2 \big/ h < -\varphi_s \gamma_s < g x_N^{\max}(t_{k+1} - t_k)^2 \big/ h \,, \\
& g y_N^{\min}(t_{k+1} - t_k)^2 \big/ h < \varphi_s \chi_s < g y_N^{\max}(t_{k+1} - t_k)^2 \big/ h \,, \\
& 0 < \frac{5g}{4r^2} \mu_{\max}(t_{k+1} - t_k)^2 \,,
\end{aligned}
$$

where $\Delta\chi$, $\Delta\gamma$, and $\Delta\varphi$ in Eq. (6) have all been replaced with values of zero. χ_s, γ_s, and φ_s are the stable strides, which will be equal to the commanded strides denoted by χ_c, γ_c, and φ_c when the commands conform to the stride limitations described by Eq. (8). The subsequent 3D-LIPM-based preview controller will not introduce footstep changes and will generate trajectories strictly in accordance with these pre-defined footsteps. As a result, the planned gait will exhibit no bias from the intentions once the entire system has reached a stable state.

For the former task, the comparison and conclusion are not so simple. The chief component of the MPC method is quadratic programming with linear constraints, whereas the footstep optimization defined in Eq. (9) is also of this type but on a much smaller scale (only three variables) and with nonlinear constraints. The costs of solving these two kinds of problem can be regarded as similar, but the MPC method must run at each sampling step, which will always be at a much higher frequency than that of the footsteps. Additionally, the proposed footstep adaption method can run in parallel with the COM trajectory generation on an independent processer. As a result, the proposed adaption approach seems more beneficial. However, the MPC method, as a coupled approach, does more work in a single cycle of optimization. It can handle uncertain errors or disturbances through dynamic feedback. This functionality has not yet been explored via the omnidirectional vehicle model.

Stability is the last but not the least difference between these two strategies. References [2, 9] provide substantial criteria based on ZMP theory, especially within a single support period. Therefore, the COM trajectories and adapted footsteps will fulfill the ZMP criteria at all times. By contrast, the footsteps generated via the approximation provided by the omnidirectional vehicle model are based on the IPCR, which is not as precise as the constraints in the 3D-LIPM. This is because of the

Table I. Comparison of the key parameters of the physical robot MOS used in the experiments and the model robot used in the dynamics simulations.

Parameter name/unit	Physical robot MOS	Sample model in OpenHRP
Body height/mm	583	1370
Body weight/kg	5.34	97.8
COM height/mm	344	750
Foot size/mm×mm	125×95	250×180

trade-off between the abstraction layer and the model precision. However, recall that the 3D-LIPM also differs considerably from a real humanoid robot; the stability criteria derived from this model also are not strictly valid, but are merely a much closer approximation to a physical robot than is the omnidirectional vehicle model.

6.2. Applicability of the footstep adaption strategy

Theoretically, the proposed footstep adaption strategy can be implemented on almost any humanoid robot, so long as it is capable of walking omnidirectionally. Moreover, the determination of the model constants in Section 5 is based on iterative experimental optimization and requires no assumptions other than the applicability of the 3D-LIPM to generate feasible motion. However, a physical implementation may behave differently from expectations. For example, walking smoothness is a fundamental requirement because no large oscillations or impulses from alternate foot landings are considered in the vehicle model. A humanoid is assumed to move calmly and steadily if given rational joint trajectories, even at a rapid walking speed, whereas in a physical robot, insufficient stiffness of the mechanical structure, control error of the actuators and unavoidable impulses may introduce large vibrations and randomness of motion, especially during rapid walking. These factors may cause a robot to tip over before a sharp turn such as that depicted in Fig. 1. These phenomena are more severe for an adult-sized humanoid robot than for a child-sized one because a larger robot has longer legs, a more complex structure and larger reaction forces. However, solutions to these difficulties may require mechanical design modifications and online stabilization, which are not discussed in detail in this research.

7. Simulation and Experiments

Both computer simulations and physical experiments were performed to test the proposed strategy. However, the robots used for the two types of tests were different: the simulations were performed for a normal-sized robot to eliminate the larger randomness of motion mentioned above, whereas for the physical implementation, we used a child-sized humanoid robot for better control of the walking smoothness. The black-box optimization process was conducted for both kinds of robots to obtain the best constants for each. Some of the key specifications of the robots are compared in Table I.

7.1. Simulations and discussion

OpenHRP,[36] which is open source and is specifically designed for but not limited to humanoid robotics, was chosen as the dynamics simulator. The robot model described in Table I is the sample that is included in the OpenHRP wrapper. It was designed to walk at 0.4 s/step. The simulations were run on a computer with an Intel Core i7 CPU and 2GB of memory with Ubuntu 10.04 installed. A single standard walking experiment is depicted in Fig. 7.

We implemented the online footstep adaptation algorithm using NLOPT,[37] an open-source collection of optimizers. Specifically, the Augmented Lagrange Method,[30] a penalty-function-based approach, was used. The SURROGATES[38] and DACE[39] toolboxes for MATLAB were used for Latin hypercube experiments, Kriging modeling and optimization. The initial values of the omnidirectional vehicle constants, which were rather conservative, were tuned by hand and are shown in Table II. The results of surrogate-model-based optimization, consisting of initial 49 experiments and 120 subsequent iterative walking experiments, are also shown in Table II. The values for a failed experiment, in which the robot fell over, are also listed in the last column for comparison.

Table II. Manually tuned values and optimization results from 49 Latin hypercube experiments and 120 iterative experiments.

Name/unit	Manually tuned #1	Manually tuned #2	Optimization result #1	Optimization result #2	Failed result
k_1	−1.6	−0.2	−0.20	−0.69	−1.0
k_2	1.6	0.2	0.87	0.81	1.0
k_3	1.6	0.2	0.84	0.83	1.0
k_5	1.0	0.2	0.97	0.96	1.0
χ_{max}/mm	100	270	192	175	175
γ_{max}/mm	40	80	68	66	66
$\varphi_{max}/^\circ$	10	20	20	20	20
Score	0.427	0.408	0.186	0.183	$+\infty$

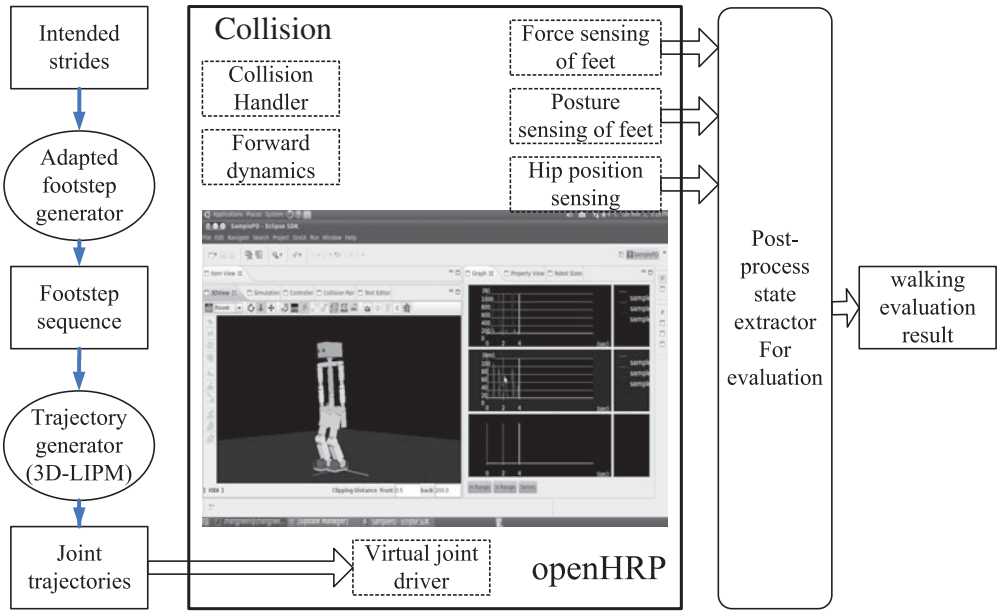


Fig. 7. Overview of the simulations performed using the OpenHRP platform.

We observe from Table II that the manually tuned constants achieved values of the tracking error that were larger than 0.4, whereas the two best results achieved scores below 0.2. Therefore, the proposed optimization approach greatly enhances the reactive omnidirectional walking quality that can be achieved while imposing an acceptable experimental workload (i.e., fewer than 200 trials).

The target velocity tracking results corresponding to optimization result #1 in Table II are shown in Fig. 8. The footsteps and ZMP positions during walking are depicted in Fig. 9.

We can observe from Fig. 8 that the robot gradually sped up as it started to walk and then slowed again before back-stepping, turning, or sidestepping, with reasonable acceleration and deceleration. The planned footstep sequence was converted into a stable walking gait and successfully executed by the model robot without falling over. The ZMP positions, though approaching the borders of the feet, are all within the support polygon, especially during the period of abrupt changes in walking intention. This outcome indicates that the robot could properly handle the inter-pace adjustments and foot relocations by solving the optimization problem defined by Eq. (9) to smooth the sharp changes in speed intentions and guarantee stability.

For comparison, the footsteps and ZMP positions from the failed experiment presented in Table II are depicted in Fig. 10. The configuration shares the same maximum stride as in optimization result #2 but includes different values of k_1, k_2, and k_3, which mean that the robot does not adapt its footsteps but instead reaches its walking intentions in a single step.

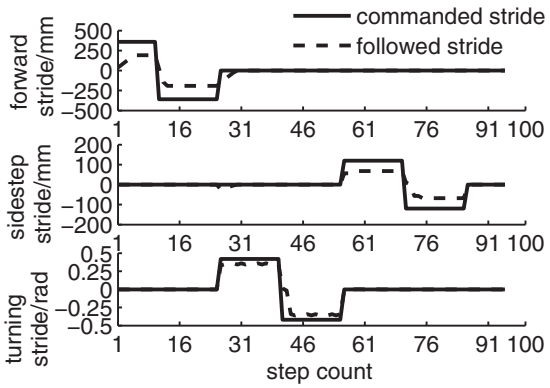
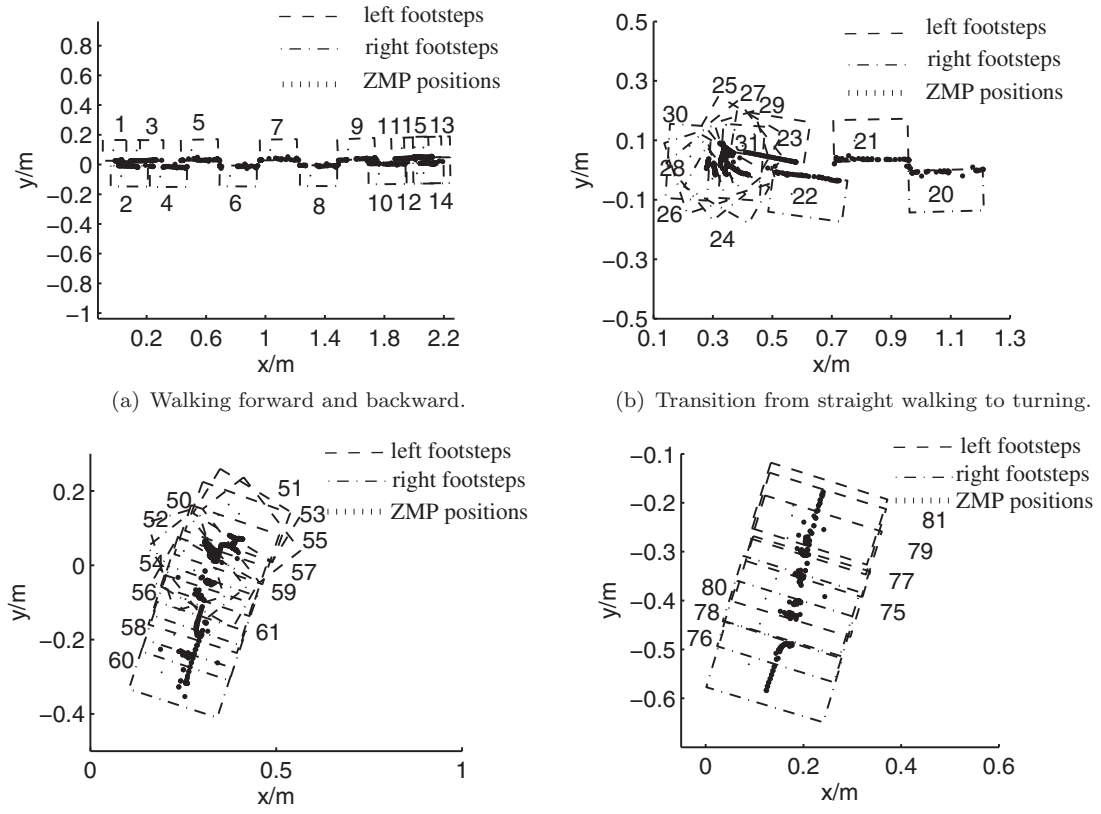


Fig. 8. Footstep adaptation results in a standard experiment with the optimized model constants.



(a) Walking forward and backward.

(b) Transition from straight walking to turning.

(c) Transition from turning to sidestepping.

(d) Transition from sidestepping to stopped.

Fig. 9. Footsteps and the corresponding ZMP positions in a standard experiment with the optimized model constants. (a) Walking forward and backward. (b) Transition from straight walking to turning. (c) Transition from turning to sidestepping. (d) Transition from sidestepping to stopped.

Consequently, it cannot complete the walking experiment because the ZMP positions fall outside of the supporting polygon. This result illustrates the necessity of foot location adaptation during reactive walking.

An arbitrary command sequence was then issued after the standard experiments were performed, and the robot was able to follow it while maintaining walking stability, as illustrated in Fig. 11. This result demonstrates that the constants reported in Table II are suitable not only for the standard experiment but also for general reactive omnidirectional walking.

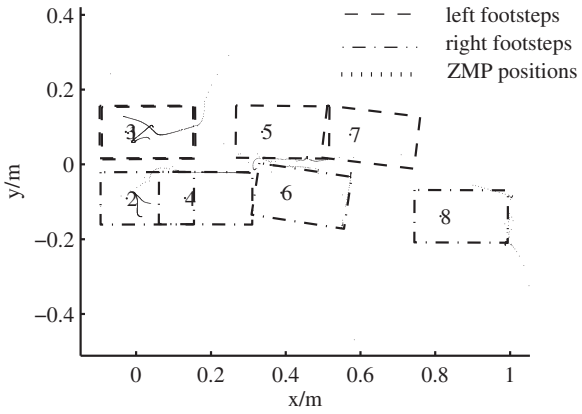


Fig. 10. Footsteps and the corresponding ZMP positions in a failed experiment.

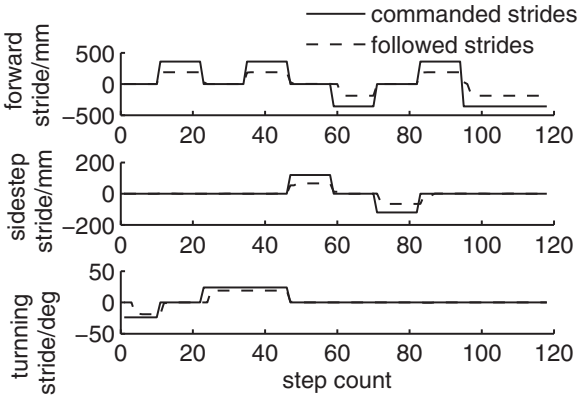


Fig. 11. The tracking results for an arbitrary stride command sequence using the black-box-optimized constants.

7.2. Deployment on a physical humanoid robot

The entire scheme was also tested on a physical robot named MOS (Master Of Soccer; see the left-hand side of Fig. 12), whose forebear participated in the humanoid league of the RoboCup from 2012 to 2014, operated by team TH-MOS[40] and team IKid.[41] It is equipped with a two-level control system, as shown in Fig. 12 (right): the upper onboard computer handles decision-making, footstep generation, and hip-foot trajectories, whereas the lower controller is responsible for inverse kinematics calculations and the dispatch of joint commands in real time. The procedure used to record and optimizing the model constants was the same as in the simulations, except that the experiments were performed directly on MOS.

The generation of footsteps via optimization as defined in Eq. (9) can be completed in less than 100 ms, which is only 1/3 of the duration of a single step and therefore satisfies the real-time requirement.

With its own optimized constants, MOS was issued omnidirectional commands as walking intentions; the snapshots of the walking sequence shown in Fig. 13 correspond to the footstep sequence shown in Fig. 14.

As observed from these snapshots and the stride-following footsteps, the robot was able to track the issued walking intentions to the greatest possible extent, and the sharp changes in the commands were smoothed, just as in the simulations. While walking, the robot gradually transitioned from back-stepping to leftward turning and then to forward walking. The online stride adaptation strategy was therefore successfully verified on this small robot with a limited computational capacity.

Although a set of model constants was determined using the same procedure as in the simulations, the approach does not satisfy the assumption of Eq. (15) because a physical experiment lacks determinacy. Even given the same model constants, the performance of a real robot tends to be variable, and the optimization does not converge if the experiment conditions are not carefully controlled. Fortunately, this shortcoming does not invalidate the applicability of the strategy as a

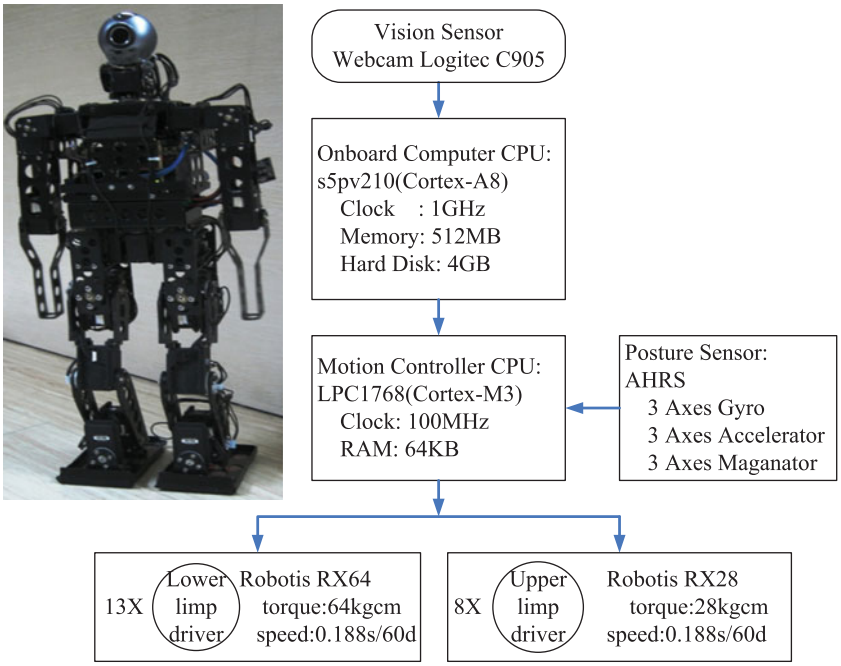


Fig. 12. Appearance of the humanoid robot MOS and its control system.

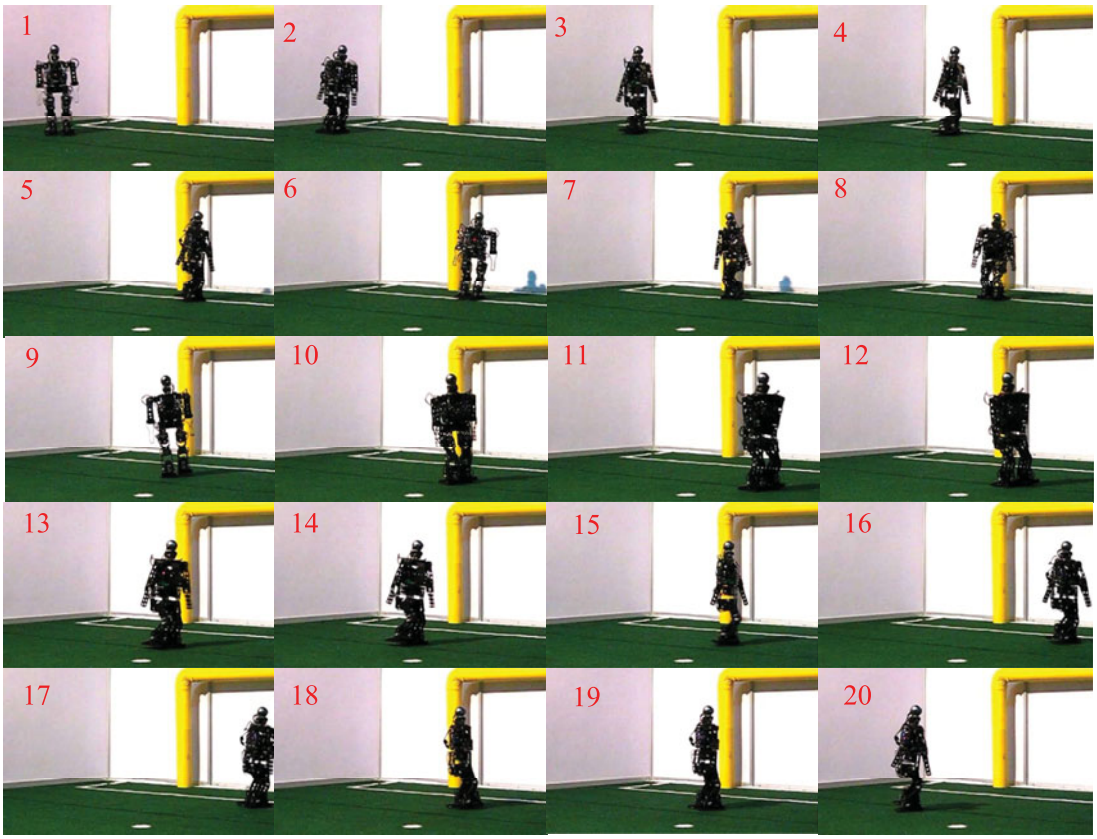


Fig. 13. Snapshots of MOS following the omnidirectional walking intentions in physical robot experiments.

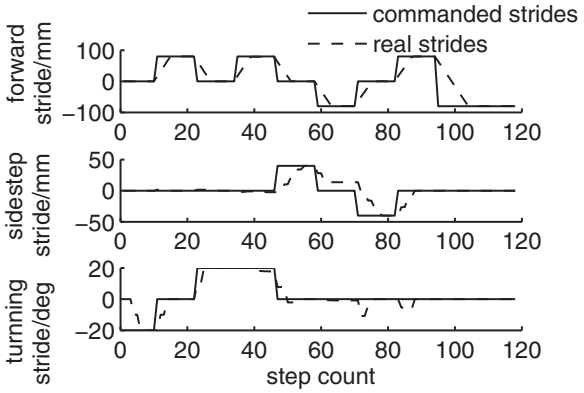


Fig. 14. Footstep sequence following the omnidirectional walking commands issued to MOS.

whole because there are already several solutions available for noisy systems of this type, such as that proposed in ref. [42], and these methods will be tested on a real robot in future research.

8. Conclusion and Future Work

In this paper, we present a foot placement strategy for omnidirectional reactive walking following arbitrary intentions, representing the following contributions to the field:

A hierarchical planning framework is proposed that separates the footstep adaptation task from that of COM trajectory generation. Thus, the time interval for footstep planning is increased to one stride, and the heavy computational load is relieved. The planner can be implemented on a physical robot with quite limited computing power.

A novel omnidirectional moving vehicle model is adopted to analytically describe the IPCR in terms of inequalities. Through nonlinear optimization based on these inequalities and other kinematic constraints, it is possible to generate the foot locations that best follow the issued commands while eliminating the unnecessary bias from the desired strides that arises in the coupled approach.

By solving a black-box optimization problem based on the mapping between the evaluation of a standard walking experiment and the seven effective model constants, it is possible to avoid complex derivations of the unique characteristics of humanoid walking dynamics, and the surrogate-model-based approach can be adopted to reduce the necessary number of experiments. The constants determined using this method do not function only for the standard experiments from which they are obtained but rather are equally applicable to arbitrary walking intentions.

In this paper, only large variations in the issued commands were addressed using the proposed methods; the problem of stride adaptations necessitated by disturbances such as external forces or uneven terrain remains uninvestigated. Future work may focus on a unified strategy to cope with all of these situations. In addition, the performance gap between a computer simulation and a physical robot with regard to surrogate-model-based optimization is also worthy of further investigation.

Acknowledgments

The authors would like to thank the staff of the Robotics and Automation Laboratory of the Mechanical Engineering Department of Tsinghua University for their support. We also gratefully acknowledge the financial support from the National Natural Science Foundation of China under Grants 61403225 and 51305436 and the Fundamental Research Program of Shenzhen under Grant JCYJ20140901003939038. The authors are also grateful to every member who has ever participated in the TH-MOS team; this study would not have been possible without the development of the robot platform and the many fruitful discussions with all members of that team.

References

1. J. Chestnutt *et al.* "An Intelligent Joystick for Biped Control," *Proceedings of the IEEE International Conference on Robotics and Automation*, Piscataway, NJ (May 15–19, 2006) pp. 860–865.

2. C. Dune *et al.,* "Cancelling the Sway Motion of Dynamic Walking in Visual Servoing," *Proceedings of IEEE International Conference on Intelligent Robots and Systems*, Piscataway, NJ (Oct. 18–22, 2010) pp. 3175–3180.
3. S. Kajita *et al.,* "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*, Piscataway, NJ, Vol. 1 (Oct. 29–Nov. 03, 2001) pp. 239–246.
4. S. Kajita *et al.* "Biped Walking Pattern Generation by Using Preview Control of Zero-Moment Point," *Proceedings of IEEE International Conference on Robotics and Automation*, Piscataway, NJ, Vol. 2 (Sep. 14–19, 2003) pp. 1620–1626.
5. S. Czarnetzki, S. Kerner and O. Urbann, "Observer-based dynamic walking control for biped robots," *Robot. Auton. Syst.* **57**, 839–845 (2009).
6. X. Zeyang, X. Jing and C. Ken, "Global navigation for humanoid robots using sampling-based footstep planners," *IEEE/ASME Trans. Mechatronics* **16**, 716–723 (2011).
7. X. Zeyang, X. Jing and C. Ken, "Parameter self-adaptation in biped navigation employing nonuniform randomized footstep planner," *Robotica* **28**, 929–936 (2010).
8. M. Morisawa *et al.,* "Motion Planning of Emergency Stop for Humanoid Robot by State Space Approach," *Proceedings of IEEE International Conference on Intelligent Robots and Systems*, Piscataway, NJ (Oct. 9–15, 2006) pp. 2986–2992.
9. A. Herdt *et al.,* "Online walking motion generation with automatic footstep placement," *Adv. Robot.* **24**, 719–737 (2010).
10. S. Piperakis, E. Orfanoudakis and M. G. Lagoudakis, "Predictive Control for Dynamic Locomotion of Real Humanoid Robots," *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems*, Piscataway, NJ (Sep. 14–18, 2014) pp. 4036–4043.
11. D. Dimitrov, A. Paolillo and P. B. Wieber, "Walking Motion Generation with Online Foot Position Adaptation Based on $l-1$ and $l-\infty$ Norm Penalty Formulations," *Proceedings of IEEE International Conference on Robotics and Automation*, Piscataway, NJ (May 9–13, 2011) pp. 3523–3529.
12. J. Zhang, L. Liu, C. Li and K. Chen, "Parametric omnidirectional gait planning of humanoid robots(in Chinese)," *Robotica* **36,** 210–217 (2014).
13. H. Qiang *et al.,* "Planning walking patterns for a biped robot," *IEEE Trans. Robot. Autom.* **17**, 280–289 (2001).
14. G. Dip, V. Prahlad and P. D. Kien, "Genetic algorithm-based optimal bipedal walking gait synthesis considering tradeoff between stability margin and speed," *Robotica* **27**, 355–365 (2009).
15. L. Hu, C. Zhou and Z. Sun, "Estimating biped gait using spline-based probability distribution function with Q-learning," *IEEE Trans. Ind. Electron.* **55**, 1444–1452 (2008).
16. H. Kensuke *et al.* "An analytical method on real-time gait planning for a humanoid robot," *Int. J. Humanoid Robot.* **3**, 1–19 (2006).
17. J. Strom, G. Slavov and E. Chown, "Omnidirectional Walking Using ZMP and Preview Control for the NAO Humanoid Robot," **In**: *RoboCup 2009: Robot Soccer World Cup XIII* (Springer, Berlin Heideiberg, 2010) pp. 378–389.
18. D. Gouaillier, C. Collette and C. Kilner, "Omnidirectional Closed-Loop Walk for NAO," *Proceedings of 2010 IEEE-RAS International Conference on Humanoid Robots*, Piscataway, NJ (Dec. 6–8, 2010) pp. 448–454.
19. N. Shafii *et al.,* "Omnidirectional Walking and Active Balance for Soccer Humanoid Robot," **In**: *Progress in Artificial Intelligence* (Springer, Berlin, Heidelberg, 2013) pp. 283–294.
20. C. Graf and R. Thomas, "A Closed-loop 3D-LIPM Gait for the RoboCup Standard Platform League Humanoid," *Proceedings of the 4th Workshop on Humanoid Soccer Robots in Conjunction with the 2009 IEEE-RAS International Conference on Humanoid Robots*, Piscataway, NJ (2009) pp. 30–37.
21. S. Song, Y. J. Ryoo and D. W. Hong "Development of An Omnidirectional Walking Engine for Full-Sized Lightweight Humanoid Robots," *Proceedings of the ASME 2011 International Design Engineering Technical Conferences and Computers and Information in Engineering Conference* (Aug. 28–31, 2011) pp. 847–854.
22. J. Alcaraz-Jimenez, D. Herrero-Perez and H. Martinez-Barbera, "Motion planning for omnidirectional dynamic gait in humanoid soccer robots. *J. Phys. Agents* **5**, 25–34 (2011).
23. P. B. Wieber, Trajectory Free Linear Model Predictive Control for Stable Walking in the Presence of Strong Perturbation," *Proceedings of IEEE International Conference on Humanoid Robots*, Piscataway, NJ (Dec. 4–6, 2006) pp. 137–142.
24. M. Vukobratovic and Y. Stepanenko, On the stability of anthropomorphic systems. *Math. Biosci.* **15**, 1–37 (1972).
25. A. Herdt, N. Perrin and P. B. Wieber, "Walking without Thinking About It," *Proceedings of 2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Piscataway, NJ (Oct. 18–22, 2010) pp. 190–195.
26. M. Garcia *et al.,* "Vision-guided motion primitives for humanoid reactive walking: Decoupled versus coupled approaches. *Int. J. Robot. Res.* **34**, 402–419 (2014).
27. D. Dimitrov, A. Sherikov and P. B Wieber, "A Sparse Model Predictive Control Formulation for Walking Motion Generation," *Proceedings of 2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, Piscataway, NJ (Sep. 25–30, 2011) pp. 2292–2299.

28. J. Pratt *et al.,* "Capture Point: A Step Toward Humanoid Push Recovery," *Proceedings of IEEE-RAS International Conference on Humanoid Robots* (2006) pp. 200–207.
29. K. Svanberg, "A class of globally convergent optimization methods based on conservative convex separable approximations," *SIAM J. Optim.* **12**, 555–573 (2002).
30. A. R. Conn, M. Gould and P. Toint, "A globally convergent augmented Lagrangian algorithm for optimization with general constraints and simple bounds," *SIAM J. Numer. Anal.* **28**, 545–572 (1991).
31. D. Orin, A. Goswami and H. Lee, "Centroidal dynamics of a humanoid robot," *Auton. Robots* **35**(2–3), 161–176 (2013).
32. T. Hemker, M. Stelzer and O. V. Stryk, "Efficient walking speed optimization of a humanoid robot," *Int. J. Robot. Res.* **28**, 303–314 (2009).
33. J. Sacks *et al.* "Design and analysis of computer experiments. *Stat. Sci.* **4**, 409–435 (1989).
34. D. R. Jones, M. Schonlau and W. J. Welch, "Efficient global optimization of expensive black-box functions," *J. Global Opt.* **13**, 455–492 (1998).
35. S. J. Bates, J. Sienz and V. V. Toropov, "Formulation of the optimal Latin hypercube design of experiments using a permutation genetic algorithm,"*AIAA J.* **2011**, 1–7 (2004).
36. F. Kanehiro, H. Hirukawa and S. Kajita, "OpenHRP: Open architecture humanoid robotics platform," *Int. J. Robot. Res.* **23**, 155–165 (2004).
37. S. G. Johnson, "The NLopt Nonlinear-Optimization Package," (2014). Available from: http://ab-initio.mit.edu/nlopt [cited Sep. 22, 2015].
38. Viana FAC, SURROGATES Toolbox User's Guide, Version 3.0 ed. Gainesville, FL, USA (2011). Available at http://sites.google.com/site/felipeacviana/surrogatestoolbox
39. S. N. Lophaven, H. B. Nielsen and J. Sndergaard, "DACE: A MATLAB Kriging Toolbox," (2007). Available at http://www.immdtu.dk/ hbn/dace/ [accessed Nov. 7, 2007].
40. X. Xiao *et al.* "Team TH-MOS," (2014). Available at http://fei.edu.br/rcs/2014/TeamDescriptionPapers/Humanoid/index.html.
41. J. Bi *et al.* "Team Description Paper for Team I-KID RoboCup 2014," (2014). Available at http://fei.edu.br/rcs/2014/TeamDescriptionPapers/Humanoid/index.html.
42. V. Picheny *et al.,* "Quantile-based optimization of noisy computer experiments with tunable precision," *Technometrics* **55**, 2–13 (2013).

Appendix A. Mathematical derivation of the IPCR inequalities from the omnidirectional vehicle model

x_N and y_N are obtained from Eq. (3) as shown in Eq. (A1):

$$\begin{aligned} x_N &= \frac{F_x h}{G}, \\ y_N &= \frac{F_y h}{G}. \end{aligned} \tag{A1}$$

By combining these expressions with Eqs. (1) and (2) and using the gravity equation $G = mg$, we obtain Eq. (A2):

$$\begin{aligned} x_N &= -\frac{(\dot{v}_x + \omega v_y)h}{g}, \\ y_N &= -\frac{(\dot{v}_y + \omega v_x)h}{g}. \end{aligned} \tag{A2}$$

g is the gravitational constant. When χ, γ, and φ are defined as the stride in the forward direction, the stride in the lateral direction, and the turning angle, respectively, Eq. (A3) below holds for (t_k, t_{k+1}), which is the time duration of a single step:

$$\begin{aligned} \chi &= \int_{t_k}^{t_{k+1}} v_x d\tau, \\ \varphi &= \int_{t_k}^{t_{k+1}} \omega d\tau. \end{aligned} \tag{A3}$$

Under the assumption of piecewise constant acceleration within a step, integration of both sides of Eq. (A2) yields

$$\begin{aligned}\int_{t_k}^{t_{k+1}} x_N d\tau &= -\int_{t_k}^{t_{k+1}} \frac{(\dot{v}_x + \omega v_y)h}{g} d\tau = -\frac{h}{g}\left(\delta v_x + \int_{t_k}^{t_{k+1}} \omega v_y d\tau\right), \\ \int_{t_k}^{t_{k+1}} y_N d\tau &= -\int_{t_k}^{t_{k+1}} \frac{(\dot{v}_y + \omega v_x)h}{g} d\tau = -\frac{h}{g}\left(\delta v_y + \int_{t_k}^{t_{k+1}} \omega v_x d\tau\right),\end{aligned} \tag{A4}$$

where δv_x and δv_y are the variations in v_x and v_y during (t_k, t_{k+1}) and are related to $\delta\chi$ and $\delta\gamma$ as follows:

$$\begin{aligned}\delta\chi &= \frac{1}{2}\delta v_x (t_{k+1} - t_k), \\ \delta\gamma &= \frac{1}{2}\delta v_y (t_{k+1} - t_k).\end{aligned} \tag{A5}$$

Because of the constant acceleration assumption, we can approximate the second term in Eq. (A4) as

$$\int_{t_k}^{t_{k+1}} \omega v_x d\tau \approx \frac{\varphi\chi}{t_{k+1} - t_k}. \tag{A6}$$

When δv_x, δv_y, and the second integration term are replaced with Eqs. (A5) and (A6), Eq. (A4) becomes

$$\begin{aligned}\int_{t_k}^{t_{k+1}} x_N d\tau &\approx -\frac{h}{g(t_{k+1} - t_k)}(2\delta\chi + \varphi\gamma), \\ \int_{t_k}^{t_{k+1}} y_N d\tau &\approx -\frac{h}{g(t_{k+1} - t_k)}(2\delta\gamma + \varphi\chi).\end{aligned} \tag{A7}$$

Because x_N and y_N are bounded in definite ranges of $(x_N^{\min}, x_N^{\max})$ and $(x_N^{\min}, x_N^{\max})$, respectively, from Eq. (4), the following can be derived:

$$\begin{aligned}x_N^{\min}(t_{k+1} - t_k) &< \int_{t_k}^{t_{k+1}} x_N d\tau < x_N^{\max}(t_{k+1} - t_k), \\ y_N^{\min}(t_{k+1} - t_k) &< \int_{t_k}^{t_{k+1}} y_N d\tau < y_N^{\max}(t_{k+1} - t_k).\end{aligned} \tag{A8}$$

Thus, we obtain the first two constraint functions in Eq. (6). For the final constraint regarding turning, M_z in Eq. (5) is replaced with $J_c\varepsilon$, and integration of both sides yields

$$\begin{aligned}&\int_{t_k}^{t_{k+1}} \mu d\tau = \int_{t_k}^{t_{k+1}} \frac{J_C\varepsilon}{G} d\tau = \frac{J_C(\omega_{k+1} - \omega_k)}{G} = \frac{2J_C\delta\varphi}{G(t_{k+1} - t_k)}, \\ &\left\| \int_{t_k}^{t_{k+1}} \mu d\tau \right\| < \mu_{\max}(t_{k+1} - t_k).\end{aligned} \tag{A9}$$

Because a spherical shape with $J_c = (2/5)mr^2$ is supposed for the centralized mass, from Eq. (A9), we ultimately obtain

$$\| \delta\varphi \| < \frac{5g}{4r^2}\mu_{\max}(t_{k+1} - t_k)^2, \tag{A10}$$

which serves as the final inequality in Eq. (6), thus concluding the procedure.