



From Walking to Running: 3D Humanoid Gait Generation via MPC

Filippo M. Smaldone, Nicola Scianca*, Leonardo Lanari and Giuseppe Oriolo

Dipartimento di Ingegneria Informatica, Automatica e Gestionale, Sapienza University of Rome, Rome, Italy

OPEN ACCESS

Edited by:

Enrico Mingo Hoffman,
Pal Robotics S.L., Spain

Reviewed by:

Yue Hu,
University of Waterloo, Canada
Johannes Engelsberger,
German Aerospace Center (DLR),
Germany
Francisco Javier Andrade Chavez,
University of Waterloo, Canada

*Correspondence:

Nicola Scianca
scianca@diag.uniroma1.it

Specialty section:

This article was submitted to
Field Robotics,
a section of the journal
Frontiers in Robotics and AI

Received: 15 February 2022

Accepted: 11 May 2022

Published: 16 August 2022

Citation:

Smaldone FM, Scianca N, Lanari L and
Oriolo G (2022) From Walking to
Running: 3D Humanoid Gait
Generation via MPC.
Front. Robot. AI 9:876613.
doi: 10.3389/frobt.2022.876613

We present a real time algorithm for humanoid 3D walking and/or running based on a Model Predictive Control (MPC) approach. The objective is to generate a stable gait that replicates a footstep plan as closely as possible, that is, a sequence of candidate footstep positions and orientations with associated timings. For each footstep, the plan also specifies an associated reference height for the Center of Mass (CoM) and whether the robot should reach the footstep by walking or running. The scheme makes use of the Variable-Height Inverted Pendulum (VH-IP) as a prediction model, generating in real time both a CoM trajectory and adapted footsteps. The VH-IP model relates the position of the CoM to that of the Zero Moment Point (ZMP); to avoid falling, the ZMP must be inside a properly defined support region (a 3D extension of the 2D support polygon) whenever the robot is in contact with the ground. The nonlinearity of the VH-IP is handled by splitting the gait generation into two consecutive stages, both requiring to solve a quadratic program. Thanks to a particular triangular structure of the VH-IP dynamics, the first stage deals with the vertical dynamics using the Ground Reaction Force (GRF) as a decision variable. Using the prediction given by the first stage, the horizontal dynamics become linear time-varying. During the flight phases, the VH-IP collapses to a free-falling mass model. The proposed formulation incorporates constraints in order to maintain physically meaningful values of the GRF, keep the ZMP in the support region during contact phases, and ensure that the adapted footsteps are kinematically realizable. Most importantly, a stability constraint is enforced on the time-varying horizontal dynamics to guarantee a bounded evolution of the CoM with respect to the ZMP. Furthermore, we show how to extend the technique in order to perform running on tilted surfaces. We also describe a simple technique that receives input high-level velocity commands and generates a footstep plan in the form required by the proposed MPC scheme. The algorithm is validated via dynamic simulations on the full-scale humanoid robot HRP-4, as well as experiments on the small-sized robot OP3.

Keywords: humanoid robot, model predictive control, legged locomotion, running, variable height inverted pendulum model

1 INTRODUCTION

Humanoid robots have become more and more advanced in the last decade. Using legged locomotion, they are in principle able to perform very complex and dynamic motions, such as those needed to traverse nonlevel terrain and realize running gaits. However, the design of online walking and running pattern generation algorithms is challenging from a control standpoint, as there are several hard requirements such as the need to always maintain dynamic balance, whether the robot is walking or running.

When walking on flat ground, balance is usually guaranteed by keeping the Zero Moment Point¹ (ZMP) within the convex hull of the contact surfaces, that is, the support polygon. To achieve real time control, a popular approach is to reduce the complexity of humanoid dynamics using the Linear Inverted Pendulum (LIP) model, obtained by assuming constant CoM height and neglecting angular momentum variations around the CoM. The linearity of the LIP model has allowed to efficiently define control schemes based on optimization, such as preview-based approaches for tracking the desired ZMP trajectory (Kajita et al., 2003), or MPC schemes that encode the balance requirement through constraints on the ZMP (Wieber, 2006).

1.1 Related Work

Variation of the CoM height is a key requirement for traversing nonlevel grounds. Removing the assumption of constant CoM height in the LIP leads to the Variable-Height Inverted Pendulum (VH-IP) model (Koolen et al., 2016), in which height variations have the effect of modifying the natural frequency of the pendulum. The varying natural frequency can be seen as an additional input of the model, which introduces a nonlinearity that entails higher complexity in the gait generation algorithm. Another possibility is to consider it as a time-varying parameter in the dynamic equation (Herdt et al., 2012), design the vertical motion offline, and generate the horizontal trajectories with a linear time-varying MPC. Other approaches (Hu et al., 2014; Brasseur et al., 2015) aim instead at bounding the effect of the difference between the VH-IP and LIP on the ZMP, in a way to guarantee that the latter is always inside the support polygon. It is also possible to find specific trajectories for which the vertical CoM motion has LIP-like dynamics (Englsberger et al., 2013; Zamparelli et al., 2018). This approach can be very effective but it constrains the CoM trajectories that can be generated.

Regardless of the employed model, gait generation schemes must necessarily deal with the intrinsic instability of humanoid dynamics. In the LIP, this instability is reflected in the presence of an unstable mode. This component is often called Divergent Component of Motion (DCM) (Takenaka et al., 2009a), or Capture Point (Pratt et al., 2006), and plays a major role in the stabilization of gait generation schemes. In the effort to extend this concept to the VH-IP, if one follows the same approach as for the LIP, it is hard to enforce stability on the DCM itself as it will remain coupled with the CoM dynamics (Hopkins et al., 2014). However, a truly decoupling change of coordinates can be found, (Caron et al., 2019a), but it is a time-varying change of coordinates with no available closed-form expression, as it is implicitly defined by solving a nonlinear differential equation.

A separate line of research has been directed at the generation of running motions, with early work dating back several decades (Raibert, 1986). To generate a running gait, the controller must be

able to produce a variable CoM height, but it also needs to account for flight phases, in which contact with the ground is lost. As such, none of the aforementioned control schemes for 3D walking can be directly adopted to generate a running gait. Biomechanic studies on human running (Blickhan, 1989; Novacheck, 1998) have consolidated the use of the Spring Loaded Inverted Pendulum (SLIP) to model running dynamics, which has frequently been involved in the generation of running motions for legged robots (Wensing and Orin, 2013; Secer and Saranlı, 2018; Secer and Cinar, 2020). In these works, control is realized by means of a step-to-step regulation of apex states (flight states with zero vertical velocity). Xiong and Ames, (2018) derive a spring-mass model from the robot dynamics to generate hopping motions by optimizing the second-order derivative of the leg length. Winkler et al. (2018) perform offline trajectory optimization to compute dynamic motions, involving flight phases and locomotion over tilted planes. The method makes use of a convenient polynomial parameterization to efficiently optimize phase durations, foot locations, and contact forces while enforcing dynamic consistency through Centroidal Dynamics. More recently, Li et al. (2021) propose an optimization-based CoM trajectory planning using a SLIP model during stance phases for quadrupedal hopping on two legs. SLIP dynamics are highly nonlinear and thus unsuitable for predictive control unless considerably approximated.

Although the ZMP is not defined during flight phases due to the loss of contact with the ground, balance during support phases can still be enforced by means of a ZMP criterion. ZMP-based approaches for running are proposed with VH-IP modeling of the dynamics during support phases and a free-falling mass model during flight phases (Kajita et al., 2007; Takenaka et al., 2009b; Tajima et al., 2009). The vertical motion is generated independently, the ZMP trajectory is preassigned and the CoM trajectory is obtained by numerically solving the dynamics in order to meet the next step boundary conditions. Englsberger et al. (2016) generate running gaits aimed at approximating natural human behavior, using measured data and a spatially independent polynomial encoding of the CoM trajectory. More recently, Ahn and Cho, (2021) proposed an MPC formulation for executing jumping motions where the vertical trajectory is generated offline. Also, Sugihara et al. (2021) introduced another ZMP-based MPC approach to generate 3D walking and running; constraints are enforced approximately in order to obtain closed-form solutions to speed up the execution.

We recently proposed an Intrinsically Stable MPC (IS-MPC) with an explicit stability constraint embedded in the formulation (Scianca et al., 2020). This constraint is in charge of dealing with the unstable character of humanoid dynamics, by guaranteeing that the CoM trajectory is always bounded with respect to the ZMP. Since the LIP is used as a prediction model, the DCM dynamics can be decoupled by using a simple change of coordinates, and a closed-form condition on the long-term evolution can be found. Zamparelli et al. (2018) used similar ideas and proposed an IS-MPC extension that is capable of generating 3D trajectories for the CoM. This can be realized

¹Other studies use the Centroidal Moment Pivot (CMP) point (Popovic et al., 2005) instead of the ZMP for gait generation and/or push recovery, for example, Griffin et al. (2017) and Shafiee-Ashtiani et al. (2017). The CMP purely encodes linear forces and thus does not require to assume the derivative of the centroidal angular momentum to be zero.

by constraining the allowed CoM evolution to a specific set of trajectories that present LIP-like dynamics along all three axes. While this approach is very effective in many circumstances, it limits the number of trajectories that can be generated and it is thus not applicable to every situation. Furthermore, it cannot be extended to generate running motions.

1.2 Contributions

In this article, we present a unified MPC algorithm that generates online both 3D walking and running gaits as well as transitions between them. Motions are generated in such a way to follow as closely as possible a footstep plan, in an environment constituted by horizontal patches at different heights, known as world of stairs.

The main contributions of this article can be summarized by the following points:

- we describe a scheme for generating a humanoid gait with a variable CoM height using the VH-IP model. The proposed architecture uses a two-stage MPC scheme and only requires solving standard linear-quadratic QP problems by means of the fact that the natural frequency present in the horizontal dynamics can be regarded as a time-varying parameter. With respect to the recent state of the art methods, our scheme has the advantage of maintaining a linear formulation without limiting the possible vertical CoM motions by constraining the pendulum frequency (Zamparelli et al., 2018; Sugihara et al., 2021);
- by properly shaping the vertical GRF using constraints in the first MPC stage, we introduce flight phases in order to allow the scheme to generate running motions;
- we propose a variation of our stability constraint (Scianca et al., 2020) to extend its applicability to the case of linear time-varying systems, and discuss how it is included in the formulation, both in the case of walking with variable height and running. Differently from Caron et al. (2019a), the proposed constraint is linear and can be efficiently enforced in a QP;
- we show how to extend the proposed scheme in order to allow running over tilted surfaces.

The proposed approach allows to generate gaits involving stair climbing and running for traversing complex terrain configurations. Furthermore, thanks to the ability to generate a CoM trajectory with variable height, it can realize a transition from walking to crouching (Kamioka et al., 2015), for example, in order to pass below an overhanging obstacle. The capabilities of the scheme will be demonstrated in dynamic simulations, as well as experiments for the case of walking. As a supplementary contribution, we will also present a simple footstep planner that, in response to high-level velocity commands, can be used to generate the required input for the proposed scheme.

The article is organized as follows: **Section 2** describes the dynamics of balance on nonlevel ground. The VH-IP model is introduced in **Section 3**. The overall approach is sketched out in **Section 4**, and then the MPC is presented in detail in **Section 5**, while a simple footstep planner is discussed in **Section 6**.

Simulations, as well as an experiment on the ROBOTIS OP3 humanoid robot, are shown in **Section 7** and **Section 8**. In **Section 9** we will briefly discuss an extension that allows the robot to run on tilted surfaces. **Section 10** concludes the article.

2 DYNAMIC BALANCE

The ability to maintain dynamic balance is a basic requirement of any gait generation scheme. On horizontal ground, balance may be guaranteed using the ZMP criterion, which prescribes that the ZMP (the point where the horizontal component of the moment of the ground reaction forces becomes zero) should be at all times within the support polygon, that is, the convex hull of the contact surfaces.

When moving to 3D locomotion, where the contact surfaces are not necessarily coplanar, a support polygon cannot be defined. At the same time, one needs to consider that the definition of ZMP is actually satisfied along a line, and therefore this point is not necessarily located on the ground (Sardain and Bessonnet, 2004). Based on this, several authors have proposed extensions of the ZMP criterion to the 3D case; in particular, Sugihara et al. (2021) require that the ZMP belongs to a pyramid with vertex at the CoM and base defined as the convex hull of the projections of the active contact surfaces on an arbitrary plane, typically chosen as a horizontal plane below all contact surfaces.

In this article, we are going to consider as admissible ZMP region for balance the convex hull of the active contact surfaces:

$$\mathcal{Z} = \left\{ \mathbf{p} : \mathbf{p} = \sum_{j=1}^{N_c} \gamma_j \mathbf{v}_j, \text{ with } \gamma_j \geq 0, \sum_{j=1}^{N_c} \gamma_j = 1 \right\},$$

where \mathbf{p} is² the generic 3D point and \mathbf{v}_j , $j = 1, \dots, N_c$, are the vertexes of all active contact surfaces. It is easy to verify that \mathcal{Z} is a subset of the earlier-defined pyramid. In a *world of stairs*, where all contact surfaces are parallel, \mathcal{Z} becomes an *oblique prism* in double support (see **Figure 1**, left). On flat ground, \mathcal{Z} is simply reduced to the support polygon.

3 SYSTEM MODELING

In this article, we will use the VH-IP model (Caron et al., 2019a) to characterize the relationship between the CoM and the ZMP.

Under the assumption of zero angular momentum around the CoM, the Newton and Euler (with respect to the ZMP) equations for the humanoid lead to

$$m(\ddot{\mathbf{p}}_c - \mathbf{g}) = \mathbf{f}, \quad (1)$$

$$(\mathbf{p}_z - \mathbf{p}_c) \times \mathbf{f} = 0, \quad (2)$$

where m is the robot mass, $\mathbf{p}_c = (x_c, y_c, z_c)$ is the CoM, $\mathbf{g} = (0, 0, -g)$ is the gravity acceleration, \mathbf{f} is the ground reaction force (GRF in the following) and $\mathbf{p}_z = (x_z, y_z, z_z)$ is the ZMP (see **Figure 2**).

²Throughout the article, points are represented by their coordinate vectors.

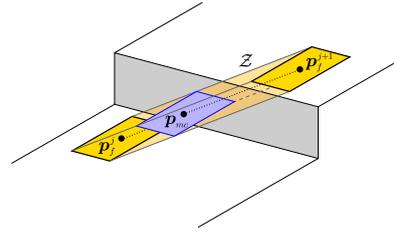
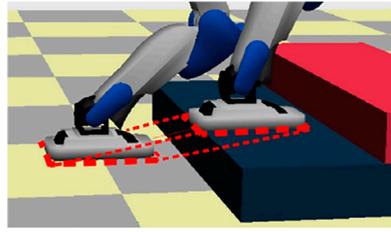


FIGURE 1 | Left: In a world of stairs, where all contact surfaces are parallel to the ground, the admissible ZMP region \mathcal{Z} in double support is an oblique prism. Right: The moving constraint region (blue) is a horizontal slice of \mathcal{Z} (yellow) that slides between two consecutive footsteps (see **Section 5.1**).

The CoM dynamics are deduced from **Eqs (1, 2)**. In particular, the vertical CoM dynamics is directly given by the third equation in **(1)**, namely

$$\ddot{z}_c = \frac{f_z}{m} - g, \quad (3)$$

where f_z is the vertical component of the GRF. The horizontal dynamics are instead obtained as:

$$\ddot{x}_c = \lambda(x_c - x_z) \quad (4)$$

$$\ddot{y}_c = \lambda(y_c - y_z), \quad (5)$$

where

$$\lambda = \frac{\ddot{z}_c + g}{z_c - z_z} = \frac{f_z}{m(z_c - z_z)} \quad (6)$$

represents the (variable) squared natural frequency of the pendulum.

Together, **Eqs 3–5** constitute the dynamic model of our system. We consider as control inputs the horizontal ZMP components x_z, y_z , and the vertical GRF f_z , while the profile of the vertical ZMP component z_z is assumed to be uniquely determined³ once the footstep plan is specified (see **Section 5.1**).

Note the triangular structure of our model: the vertical dynamics **(3)** are independent of the horizontal dynamics **(4–5)**. Once f_z is chosen, the evolution of z_c is determined, and the same is true for λ ; at this point, the horizontal dynamics can be interpreted as a linear time-varying system. We will exploit this fact when designing our MPC controller.

The assumption that f_z is a control input requires that at least one contact with the ground is active. During flight phases, which are an integral part of a running gait, the GRF is identically zero. Setting $f = \mathbf{0}$ in **(1)** gives

$$\ddot{p}_c = g, \quad (7)$$

indicating that the motion of the CoM is entirely controlled by gravity. Note that **Eq. 7** coincides with **(3–5)** when $f_z = 0$.

³An alternative (Zamparelli et al., 2018) is to choose z_z as a control input in such a way that λ is constant. The approach adopted in this article is less constraining in terms of the CoM trajectories.

4 PROPOSED APPROACH

For a humanoid robot moving in a world of stairs, we wish to design a control scheme for replicating as closely as possible an assigned footstep plan. In general, this plan will consist of a sequence of candidate footstep poses (3D positions and orientations) with specified timings. In addition, the plan also specifies for each step a reference value for the CoM height relative to the ZMP, and whether the step itself should be performed by walking or running. A simple algorithm to compute such a footstep plan will be discussed in **Section 6**.

A block scheme of the proposed approach is shown in **Figure 3**. The footstep plan is the input to the IS-MPC (Intrinsically Stable MPC) block, which consists of two sequential stages. The first is a quadratic program (QP- z) where f_z in **(3)** is chosen over a certain control horizon to generate a vertical CoM trajectory which tracks the reference height as closely as possible. As a byproduct, the corresponding values of λ can be computed via **(6)** thanks to the triangular structure of the VH-IP model. At this point, it is possible to enter the second stage, which is another quadratic program (QP- xy) over the same control horizon, where x_z and y_z in **(4–5)** are chosen and the footsteps are adapted to generate a horizontal CoM trajectory x_c, y_c resulting in a gait which is both dynamically balanced and internally stable. A qualitative representation of the trajectories produced by each stage is shown in **Figure 4**, with reference to a typical example.

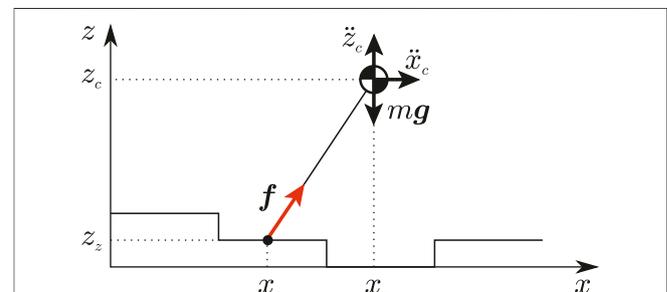


FIGURE 2 | The VH-IP model in the x - z plane. Forces acting on the robot are shown in red.

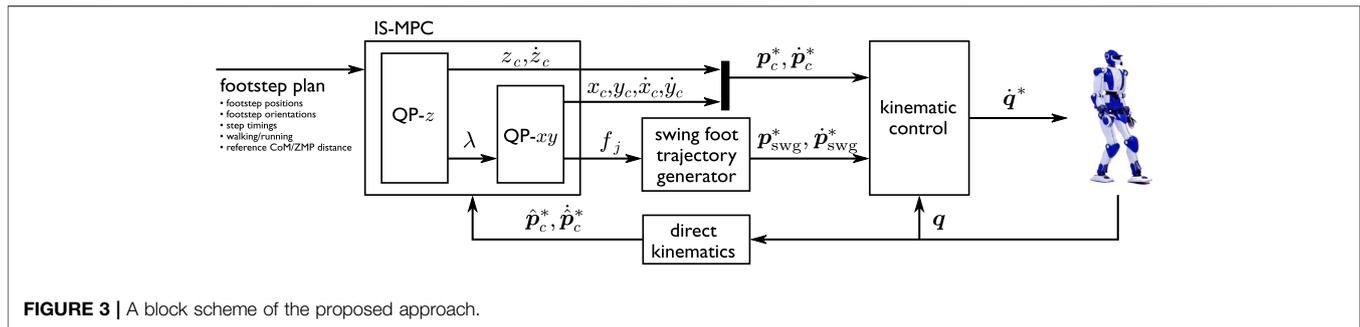


FIGURE 3 | A block scheme of the proposed approach.

The generated 3D CoM trajectory, along with an appropriate trajectory of the swing foot⁴, is sent to a kinematic control block where joint commands are computed using standard pseudoinversion techniques.

5 IS-MPC FOR 3D WALKING AND RUNNING

In this section, we will describe in detail the IS-MPC scheme for 3D walking and running based on the VH-IP model of **Section 3**.

The scheme works in discrete time over time intervals of duration δ . A sampled variable at t_k will be denoted by the k superscript, for example, $p_z(t_k) = p_z^k$. The control inputs of model (3–5), namely the horizontal components of the ZMP (x_z, y_z) and the vertical GRF f_z , are assumed to be constant over the duration of a single interval, that is, $x_z(t) = x_z^k, y_z(t) = y_z^k, f_z(t) = f_z^k$ for $t \in [t_k, t_{k+1})$.

To simplify the exposition, we will assume that all footsteps in the plan have the same orientation, taken without loss of generality to be the same of the x axis. This has the effect of decoupling the motion along the sagittal and coronal plane, making it possible to present the method for the x component only, with the understanding that a similar reasoning holds for the y component, except when specified. However, it is relatively easy to extend the proposed method to the case of variable orientation. When the orientation of the footsteps is not constant, the x and y components are coupled by a rotation matrix. As a consequence, a single QP problem involving both x and y components of the ZMP must be formulated, which however remains linear-quadratic provided that the orientation is not a decision variable in the QP, as shown in (Scianca et al., 2020).

At each time t_k , the IS-MPC block receives the current footstep plan, that is a sequence of footstep timings (t_s^1, \dots, t_s^F) and positions ($\hat{x}_f^j, \hat{y}_f^j, z_f^j$), for $j = 1, \dots, F$, over a *preview horizon* of duration $T_p = P \cdot \delta$ (see **Figure 5**). Each footstep is accompanied by a reference height h^j of the CoM relative to the ZMP, as well as information on whether the robot must reach the next footstep by walking or running. Note that the

horizontal footstep coordinates (\hat{x}_f^j, \hat{y}_f^j) are actually candidates which may be adapted by IS-MPC, while the vertical coordinates z_f^j cannot be changed, each being the height of the corresponding ground patch. The generic step is composed by a single support phase, starting at t_s^j and having duration T_{ss}^j , followed by a double support or flight phase (depending on whether the plan specifies walking or running) for the remaining duration $t_{j+1} - t_j - T_{ss}^j$. The number of footsteps F in the preview horizon will depend on the duration of each step.

On the basis of the current footstep plan, IS-MPC computes a 3D CoM trajectory and an adapted set of footsteps over a *control horizon*⁵ $T_c = C \cdot \delta$, with $T_c \leq T_p$ (see **Figure 5**). As mentioned in the previous section, this is obtained via two sequential stages, QP- z and QP- xy . Both these programs hinge upon the definition of a suitable ZMP constraint for balance, which will therefore be introduced next.

5.1 Moving ZMP Constraint

As discussed in **Section 2**, the robot is dynamically balanced if the ZMP is inside the convex hull \mathcal{Z} of the contact surfaces, which is an oblique prism in our world of stairs. Expressing this constraint in double support would involve products of decision variables, thus resulting in a nonlinear constraint. In order to preserve linearity, we employ a *moving constraint* (Abouondonia et al., 2017): the ZMP must belong to a fixed-shape region (the footprint) which slides between consecutive footsteps during double support and coincides with a footstep during single support. In other words, this region is a horizontal slice of the prism, see **Figure 1**.

The center of the moving constraint region is denoted by $p_{mc} = (x_{mc} \ y_{mc} \ z_{mc})^T$ and is expressed as a linear function of the footstep positions:

⁵Having two different horizons for planning and control is a common occurrence, as the window over which the planner operates is generally independent of the window over which the MPC optimizes. In fact, the first can range from a few steps up to the total duration of the locomotion task, while the second is usually much shorter in order to make the problem computationally tractable online. Furthermore, as shown in Scianca et al. (2020), having $T_c < T_p$ is necessary to achieve recursive feasibility, as the information in the preview horizon is used to mitigate the effect of the limited control horizon on the stability of the generated trajectories.

⁴For compactness, we do not discuss in detail the module which provides the swing foot trajectory. It uses polynomial functions whose endpoints are adjusted in order to match the footstep positions, for example, see (Scianca et al., 2017).

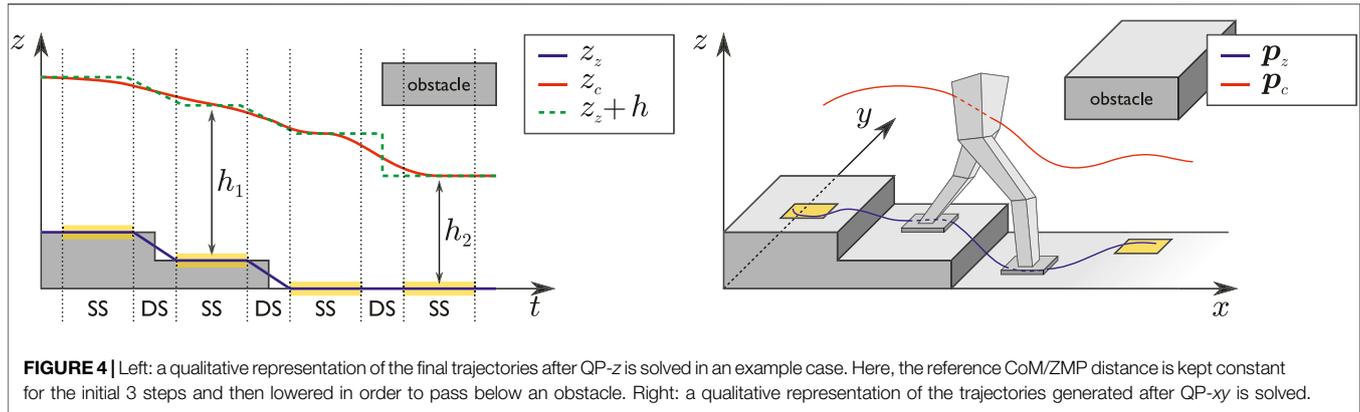


FIGURE 4 | Left: a qualitative representation of the final trajectories after QP-z is solved in an example case. Here, the reference CoM/ZMP distance is kept constant for the initial 3 steps and then lowered in order to pass below an obstacle. Right: a qualitative representation of the trajectories generated after QP-xy is solved.

$$\mathbf{p}_{mc}(t) = \sum_{j=0}^F \alpha^j(t) \mathbf{p}_f^j, \quad (8)$$

where the $\alpha^j(t)$'s are piecewise-linear functions of time, ranging from 0 and 1 and such that $\sum_{j=0}^F \alpha^j(t) = 1$, and $\mathbf{p}_f^j = (x_f^j, y_f^j, z_f^j)^T$ denotes the position of the j -th footstep. For the sake of compactness, here, we omit the expression of $\alpha^j(t)$, but we report a visual representation in **Figure 6**.

Note that, in our formulation, the moving constraint region is planar. Thus, the vertical coordinate z_z of the ZMP must at all times coincide with z_{mc} , which is uniquely determined based on the footstep plan. In fact, **Eq. 8** entails that z_{mc} only depends on the vertical coordinates of the footsteps, which are fixed and cannot be modified by IS-MPC.

5.2 Vertical QP

The goal of QP-z is to generate the vertical CoM trajectory. The prediction model is given by (3), with the decision variable f_z being piecewise-constant over intervals of duration δ :

$$\ddot{z}_c = \frac{f_z^{k+i}}{m} - g \quad \text{for } t \in [t_{k+i}, t_{k+i+1}). \quad i = 0, \dots, C-1.$$

In order to avoid slipping while walking, the GRF must satisfy a condition of sufficient friction, commonly expressed through a friction cone (Ahn and Cho, 2021). Enforcing this directly as a constraint, however, would introduce a nonlinear coupling between the vertical GRF and the horizontal CoM

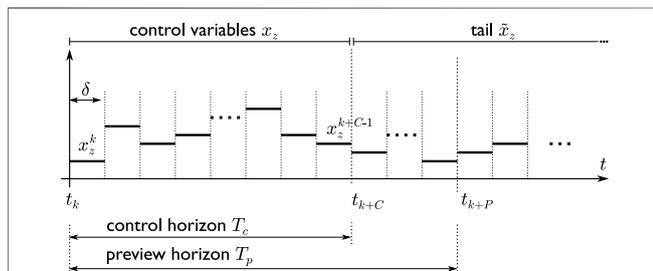


FIGURE 5 | A schematic visualizing the time discretization as well as the two horizons used in the scheme.

accelerations. To avoid this, we impose the simplified constraint

$$f_z^{k+i} \geq f_z^{\min}, \quad (9)$$

where f_z^{\min} is a value of the vertical GRF that provides sufficient friction to avoid slipping. During single support, this can be computed by using the following relationship based on a simple Coulomb friction model:

$$f_z^{\min} = \frac{m}{\mu} a_{\max},$$

where μ is the friction coefficient and $a_{\max} = \max \sqrt{\dot{x}_c^2 + \dot{y}_c^2}$ is the maximum horizontal acceleration of the CoM, which can be computed in a simulated environment over a number of trials. In practice, it is observed that this value is also appropriate in double support phases.

In correspondence of flight phases (the robot is running), constraint (9) is replaced by

$$f_z^{k+i} = 0 \quad (10)$$

in compliance with model (7).

Collect all decision variables in a vector

$$\mathbf{F}_z^k = (f_z^{k+1} \dots f_z^{k+C-1})^T$$

along with the predicted state variables

$$\dot{\mathbf{z}}_c^k = (z_c^k \dots z_c^{k+C-1})^T \quad \mathbf{z}_c^k = (z_c^k \dots z_c^{k+C-1})^T$$

and reference values for the absolute CoM height

$$\mathbf{z}_c^{k*} = (z_z^k + h^k \dots z_z^{k+C-1} + h^{k+C-1})^T. \quad (11)$$

In the latter, z_z^{k+i} is the vertical coordinate z_{mc} of the center of the moving constraint region (see the end of **Section 5.1**), while h^{k+i} is the reference CoM height relative to the ZMP as specified by the current footstep plan.

QP-z is formulated as

$$\begin{cases} \min_{\mathbf{F}_z^k} \|\mathbf{z}_c^k - \mathbf{z}_c^{k*}\|^2 + \alpha_z \|\dot{\mathbf{z}}_c^k\|^2 + \beta_z \|\Delta F_z^k\|^2 \\ \text{subject to:} \\ \bullet \text{ constraint (9), for each interval } [t_{k+i}, t_{k+i+1}] \text{ where the robot is walking} \\ \bullet \text{ constraint (10), for each interval } [t_{k+i}, t_{k+i+1}] \text{ where the robot is running,} \end{cases}$$

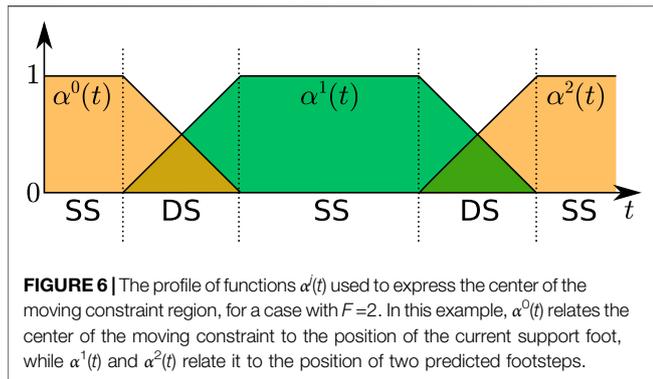


FIGURE 6 | The profile of functions $\alpha^i(t)$ used to express the center of the moving constraint region, for a case with $F=2$. In this example, $\alpha^0(t)$ relates the center of the moving constraint to the position of the current support foot, while $\alpha^1(t)$ and $\alpha^2(t)$ relate it to the position of two predicted footsteps.

where α_z, β_z are positive weights. The first term of the cost function takes care of tracking the reference CoM height, while the second term penalizes sudden height variations. The last term is included to reduce the difference between consecutive samples of the vertical GRF, so as to generate a smoother GRF profile.

As discussed in **Section 4**, the vertical GRF produced in output by QP-z is not directly applied to the system. Instead, the associated vertical CoM trajectory will be sent to the kinematic controller and converted to suitable joint commands for the robot.

5.3 Horizontal QP

The goal of QP-xy is to complete the generation of the CoM trajectory with the horizontal components. Thanks to the assumption that all footsteps have the same orientation, QP-xy itself can be split into two decoupled programs QP-x and QP-y. We will present in detail the formulation of QP-x, with the understanding that QP-y is formally identical except when explicitly noted.

When the robot is walking, its CoM obeys the horizontal dynamics (4–5), where the time-varying λ should be computed via (6) using the vertical CoM trajectory generated by QP-z. In particular, we can sample-and-hold $\lambda(t)$ to obtain a piecewise-time-invariant prediction model:

$$\ddot{x}_c = \lambda^{k+i} (x_c - x_z) \quad \text{for } t \in [t_{k+i}, t_{k+i+1}), \quad i = 0, \dots, C-1, \quad (12)$$

where

$$\lambda^{k+i} = \frac{\dot{z}_c^{k+i} + g}{z_c^{k+i} - z_z^{k+i}}.$$

In time intervals where the robot is running, its CoM obeys the flight-phase dynamics (7), and thus the prediction model becomes

$$\ddot{x}_c = 0,$$

which, as already noticed, is equivalent to setting $\lambda^{k+i} = 0$ in (12).

QP-x will enforce five kinds of constraints, namely: ZMP constraints, ground patch constraints, footstep kinematic constraints, swing foot constraints, and finally a special stability constraint.

5.3.1 ZMP Constraints

Assume⁶ the footprints to be rectangular regions with dimensions $d_{z,x}$ and $d_{z,y}$. Following the discussion in **Section 5.1**, in a generic time interval we guarantee balance by confining the ZMP to a moving constraint region having the same shape:

$$-\frac{1}{2} \begin{pmatrix} d_{z,x} \\ d_{z,y} \end{pmatrix} \leq \begin{pmatrix} x_z^{k+i} \\ y_z^{k+i} \end{pmatrix} - \begin{pmatrix} x_{mc}^{k+i} \\ y_{mc}^{k+i} \end{pmatrix} \leq \frac{1}{2} \begin{pmatrix} d_{z,x} \\ d_{z,y} \end{pmatrix}, \quad (13)$$

where the coordinates (x_{mc}, y_{mc}) of the moving constraint region are given by (8).

5.3.2 Ground Patch Constraint

To maintain linearity of the constraints, we do not allow footstep adaptations that would move the footstep to a ground patch located at a different height from that of the candidate footsteps. For this reason, we enforce the following constraint:

$$(x_f^j, y_f^j) \in \mathcal{P}_{\text{conv}}(\hat{x}_f^j, \hat{y}_f^j) \ominus \mathcal{P}_{\text{foot}}, \quad (14)$$

where $\mathcal{P}_{\text{conv}}(x, y)$ is a convex region that includes point (x, y) and is contained in a single ground patch, \ominus denotes a Minkowski difference (Montejano, 1996), and $\mathcal{P}_{\text{foot}} = \{(x, y): |x| \leq d_{z,x}/2, |y| \leq d_{z,y}/2\}$ is a rectangular region having the same dimensions of the footprint, see **Figure 7**. It is clearly beneficial to choose $\mathcal{P}_{\text{conv}}(x, y)$ as large as possible.

In this article, we assume the full knowledge of the environment and therefore the above computation can also be performed offline. For algorithmic solutions to this problem see, for example, (Deits and Tedrake, 2014; Deits and Tedrake, 2015).

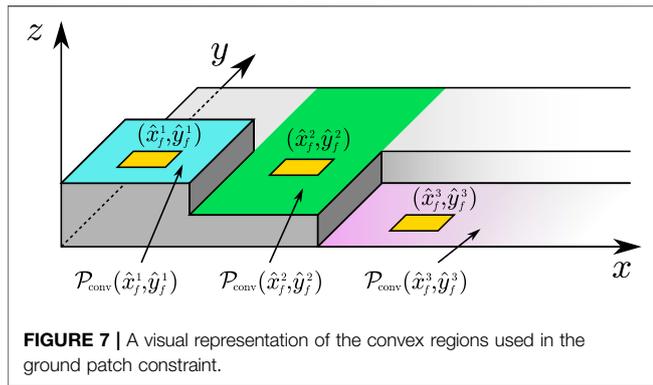
5.3.3 Kinematic Constraints

The third type of constraint ensures that footsteps are placed so as to be kinematically realizable by the robot. This means enforcing a box constraint on the step length:

$$|x_f^j - x_f^{j-1}| \leq \frac{d_{a,x}(z_f^{j-1}, z_f^j)}{2}, \quad |y_f^j - y_f^{j-1} \pm \ell| \leq \frac{d_{a,y}(z_f^{j-1}, z_f^j)}{2}, \quad (15)$$

with $d_{a,x}(z_f^{j-1}, z_f^j)$ and $d_{a,y}(z_f^{j-1}, z_f^j)$ denoting the dimensions of the admissible region along x and y , ℓ is an average coronal displacement between consecutive footsteps, and the plus/minus signs alternates between left and right footsteps. Note that $d_{a,x}(z_f^{j-1}, z_f^j)$ and $d_{a,y}(z_f^{j-1}, z_f^j)$ are assumed to be functions of z_f^{j-1} and z_f^j because they should depend on the difference in height between the two corresponding patches. Intuitively, climbing higher stairs limits the horizontal leg extension; we encode this idea by letting

⁶The choice of rectangular regions simplifies the exposition, but the reasoning can be generalized to any convex shape.



$$d_{a,x}(z_f^{j-1}, z_f^j) = \left(1 - \sigma \frac{|z_f^j - z_f^{j-1}|}{\Delta z_f^{\max}} \right) d_{a,x}^0,$$

$$d_{a,y}(z_f^{j-1}, z_f^j) = \left(1 - \sigma \frac{|z_f^j - z_f^{j-1}|}{\Delta z_f^{\max}} \right) d_{a,y}^0,$$

where $\sigma \in (0, 1)$, Δz_f^{\max} is the maximum height variation that the robot can realize, and $d_{a,x}^0, d_{a,y}^0$ are the dimensions of the admissible region on planar ground. Note that (15) is the only constraint whose expression differs between QP-x and QP-y, due to the inclusion of the lateral displacement ℓ .

5.3.4 Swing Foot Constraint

To prevent the leg joint speed from exceeding a feasible range during the swing foot motion, we enforce the following swing foot constraint (Herdt et al., 2010)

$$\left| x_{sw}^k - x_f^1 \right| \leq (t_s^0 + T_{ss}^0 - t_k) v_{sw}^{\max}, \quad (16)$$

where t_s^0 is the starting time of the current step, T_{ss}^0 the single support duration and the quantity $(t_s^0 + T_{ss}^0 - t_k)$ represents the remaining time to complete the current single support phase. This linear constraint prevents infeasible leg motions for the humanoid by increasingly limiting the distance of the next footstep x_f^1 from the current swing foot position x_{sw}^k as the foot is landing. The velocity bound term v_{sw}^{\max} can be computed from the maximum foot Cartesian velocity that the robot can realize. A similar constraint must be enforced for the y component.

5.3.5 Stability Constraint

All models of humanoid robot dynamics have an underlying unstable behavior, due to the balancing nature of these systems. As a consequence, keeping the ZMP inside the support polygon is not sufficient to generate a successful gait. In fact, a perfectly acceptable ZMP trajectory could be associated with a diverging CoM trajectory, making the resulting motion infeasible for the robot in practice. The goal of this section is to derive a constraint capable of guaranteeing the internal stability of the scheme, that is, keeping the CoM trajectory bounded with respect to the ZMP.

In previous works (Scianca et al., 2016; Scianca et al., 2020), we approached the above problem by enforcing a *stability condition* for the LIP model, in which $z_c - z_z$ is assumed to be constant. Let us consider the same model as a starting point:

$$\ddot{x}_c = \lambda_{LIP} (x_c - x_z), \quad (17)$$

where $\lambda_{LIP} = g/(z_c - z_z)$ is constant. Using the new coordinate

$$x_u = x_c + \frac{\dot{x}_c}{\sqrt{\lambda_{LIP}}}, \quad (18)$$

the unstable dynamics is isolated and expressed as:

$$\dot{x}_u = \sqrt{\lambda_{LIP}} (x_u - x_z), \quad (19)$$

with $\sqrt{\lambda_{LIP}}$ the natural frequency of the equivalent pendulum.

The instability of system (17) implies that, for a generic ZMP trajectory, x_u would diverge with respect to x_z , even if the ZMP is inside the support polygon. However, at time t_k , the system can be initialized in such a way that the free evolution of (19) exactly cancels the component of the forced evolution that would diverge with respect to x_z , allowing the CoM trajectory to remain bounded with respect to the ZMP. This special initialization depends on the future ZMP trajectory, and is identified by the following proposition.

Proposition 1. Assume that $|x_z(t') - x_z(t)| \leq a + b(t' - t)$, $\forall t' \geq t$, for some $a, b > 0$, and that

$$x_u(t_k) = \sqrt{\lambda_{LIP}} \int_{t_k}^{\infty} e^{-\sqrt{\lambda_{LIP}}(\tau - t_k)} x_z(\tau) d\tau, \quad (20)$$

then, system (17) is internally stable, that is, x_c is bounded with respect to x_z :

$$\exists M > 0: |x_c(t) - x_z(t)| \leq M, \quad t \geq t_k.$$

Proof. See **Supplementary Appendix S1**.

The first hypothesis in this proposition requires that the future ZMP trajectory is bounded by a linear function. This allows the ZMP to be discontinuous, but it does not allow arbitrarily large skips. Note that this requirement is in our case always satisfied due to the presence of the kinematic constraint. In fact, a bounding linear function for the future ZMP can easily be derived by considering the maximum allowed step size and the step duration. We omit however the details of this calculation as it is fairly tedious, and the specific expression of this bound offers no additional insight.

This result is not directly applicable to the present setting, because system (12) is time-varying and therefore the coordinate change (18) does not actually isolate the unstable dynamics. To recover a favorable situation, we will assume in our prediction model that $\lambda^{k+i} = \lambda_{LIP}$ for $t \geq t_{k+C}$, consistently with the fact that the control horizon does not extend beyond t_{k+C} . This is equivalent to defining a modified prediction model alternative to (12) given by

$$\ddot{x}_c = \begin{cases} \lambda^{k+i} (x_c - x_z) & \text{for } t \in [t_{k+i}, t_{k+i+1}), \quad i = 0, \dots, C-1 \\ \lambda_{LIP} (x_c - x_z) & \text{for } t \geq t_{k+C}, \end{cases} \quad (21)$$

with $\lambda_{\text{LIP}} = g/(x_z^{k+C} + h^{k+C})$. Note that (21) is identical to the time-varying (12) system up to the horizon t_{k+C} and becomes time-invariant after that.

For system (21) we can formulate a stability condition, as expressed by the following:

Proposition 2. Assume that $|x_z(t') - x_z(t)| \leq a + b(t' - t)$, $\forall t' \geq t$, for some $a, b > 0$, and that the current state (x_c^k, \dot{x}_c^k) satisfies

$$G \Phi(t_{k+C}, t_k) \begin{pmatrix} x_c^k \\ \dot{x}_c^k \end{pmatrix} + \int_{t_k}^{t_{k+C}} G \Phi(t_{k+C}, \tau) B(\tau) x_z(\tau) d\tau = x_u^{k+C}, \quad (22)$$

where $G = (1 \quad 1/\sqrt{\lambda_{\text{LIP}}})$, the terms $\Phi(t_{k+C}, t)$ and $B(t)$ are respectively the transition and the input matrix for system (12), and

$$x_u^{k+C} = \sqrt{\lambda_{\text{LIP}}} \int_{t_{k+C}}^{\infty} e^{-\sqrt{\lambda_{\text{LIP}}(\tau-t_{k+C})} x_z(\tau) d\tau. \quad (23)$$

Then, system (21) is internally stable, that is, x_c is bounded with respect to x_z :

$$\exists M > 0: |x_c(t) - x_z(t)| \leq M, \quad t \geq t_k.$$

Proof. System (21) is equivalent to system (17) for $t \geq t_{k+C}$. Hence, Prop. 1 states that if x_u^{k+C} is given by (23), then the subsequent evolution of x_c will be bounded with respect to x_z . On the other hand, it is $x_u^{k+C} = G(x_c^{k+C} \quad \dot{x}_c^{k+C})^T$, with

$$\begin{pmatrix} x_c^{k+C} \\ \dot{x}_c^{k+C} \end{pmatrix} = \Phi(t_{k+C}, t_k) \begin{pmatrix} x_c^k \\ \dot{x}_c^k \end{pmatrix} + \int_{t_k}^{t_{k+C}} \Phi(t_{k+C}, \tau) B(\tau) x_z(\tau) d\tau.$$

The thesis follows immediately.

Condition (22) is noncausal, as it depends on the entire future ZMP trajectory. To find a causal condition we proceed, similarly to (Scianca et al., 2020), by conjecturing an anticipative tail, that is, a ZMP trajectory $\tilde{x}_z(t)$ for $t \in [t_{k+C}, \infty)$, on the basis of the available preview information from the footstep plan. In particular, this trajectory coincides with the center of the future moving constraint region

$$\tilde{x}_z(t) = \begin{cases} x_{mc}^{k+i} & \text{for } t \in [t_{k+i}, t_{k+i+1}), \quad i = C, \dots, P-1 \\ x_{mc}^{k+P} & \text{for } t \in [t_{k+P}, \infty), \end{cases}$$

where x_{mc}^{k+i} is computed from (8). At this point, one obtains a causal version of (22) by replacing x_z with \tilde{x}_z in (23), obtaining

$$\begin{aligned} G \Phi(t_{k+C}, t_k) \begin{pmatrix} x_c^k \\ \dot{x}_c^k \end{pmatrix} + \int_{t_k}^{t_{k+C}} G \Phi(t_{k+C}, \tau) B(\tau) x_z(\tau) d\tau \\ = \sqrt{\lambda_{\text{LIP}}} \int_{t_{k+C}}^{\infty} e^{-\sqrt{\lambda_{\text{LIP}}(\tau-t_{k+C})} \tilde{x}_z(\tau) d\tau. \end{aligned} \quad (24)$$

To convert this causal condition into a constraint, it is necessary to express the integral in the left hand side in terms of the MPC decision variables, that is the samples of the piecewise-constant ZMP. This requires explicitly stating the form of the state transition matrix $\Phi(t_{k+C}, t)$ and the input matrix $B(t)$. In particular, it is sufficient to specify these terms at each sampled time instant t_{k+i} along the control horizon,

recalling that their expression will vary depending on whether t_{k+i} belongs to a support or a flight phase. The state transition matrix from t_{k+i} to t_{k+C} is given by:

$$\Phi(t_{k+C}, t_{k+i}) = \prod_{j=i}^{C-1} \Phi_{k+j} \quad i = 0, \dots, C-1,$$

where in support and flight phases we have, respectively,

$$\Phi_{k+j} = \begin{pmatrix} \cosh(\sqrt{\lambda^{k+j}} \delta) & \frac{\sinh(\sqrt{\lambda^{k+j}} \delta)}{\sqrt{\lambda^{k+j}}} \\ \sinh(\sqrt{\lambda^{k+j}} \delta) \sqrt{\lambda^{k+j}} & \cosh(\sqrt{\lambda^{k+j}} \delta) \end{pmatrix} \quad \text{and} \quad \Phi_{k+j} = \begin{pmatrix} 1 & \delta \\ 0 & 1 \end{pmatrix}.$$

Similarly, for the input matrix we have, respectively,

$$B(t_{k+i}) = \begin{pmatrix} 1 - \cosh(\sqrt{\lambda^{k+i}} \delta) \\ -\sinh(\sqrt{\lambda^{k+i}} \delta) \sqrt{\lambda^{k+i}} \end{pmatrix} \quad \text{and} \quad B(t) = \begin{pmatrix} 0 \\ 0 \end{pmatrix}$$

in support and flight phases.

Substituting these into (24) leads to the stability constraint

$$\begin{aligned} G \sum_{i=0}^{C-1} \prod_{j=i}^{C-1} \Phi_{k+j} B_{k+i} x_z^{k+i} &= \sqrt{\lambda_{\text{LIP}}} \int_{t_{k+C}}^{\infty} e^{-\sqrt{\lambda_{\text{LIP}}(\tau-t_{k+C})} \tilde{x}_z(\tau) d\tau \\ &\quad - G \prod_{i=0}^{C-1} \Phi_{k+i} \begin{pmatrix} x_c^k \\ \dot{x}_c^k \end{pmatrix}. \end{aligned}$$

5.3.6 QP-x Formulation

Collect the candidate footstep coordinates over the control horizon in vector

$$\hat{X}_f^k = (\hat{x}_f^1 \quad \dots \quad \hat{x}_f^F)^T$$

and the decision variables in vectors

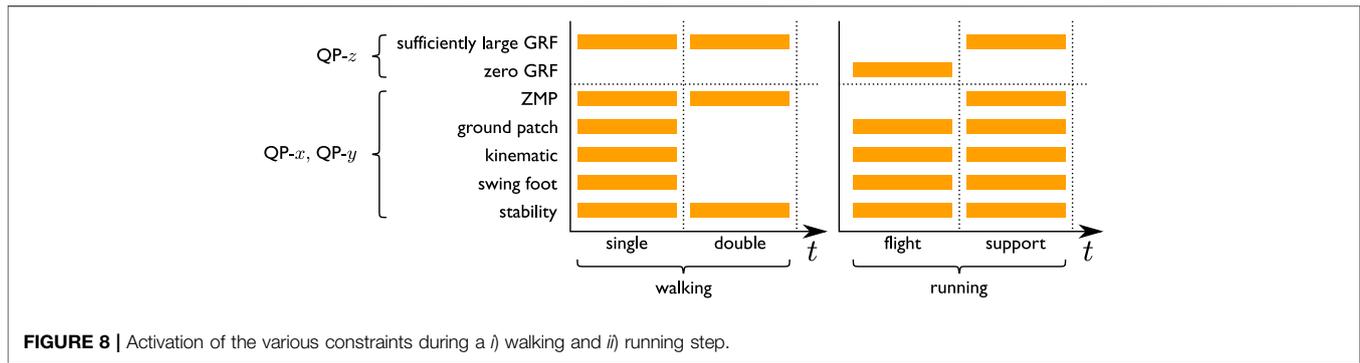
$$X_z^k = (x_z^k \quad \dots \quad x_z^{k+C-1})^T \quad X_f^k = (x_f^1 \quad \dots \quad x_f^F)^T.$$

Moreover, define $X_{mc}^k = (x_{mc}^k \quad \dots \quad x_{mc}^{k+C-1})^T$ as a vector containing the x coordinate of the center of the moving constraint region (8). Finally, define the ZMP increment vector as $\Delta X_z = (x_z^{k+1} - x_z^k \quad \dots \quad x_z^{k+C} - x_z^{k+C-1})^T$.

QP-x is formulated as:

$$\begin{cases} \min_{x_c^k, \dot{x}_c^k} \|X_z - X_{mc}^k\|^2 + \alpha_x \|\Delta X_z\|^2 + \beta_x \|X_f^k - \hat{X}_f^k\|^2 \\ \text{subject to:} \\ \bullet \text{ ZMP constraints (13, } x \text{ component), for } j = 0, \dots, F \\ \bullet \text{ ground patch constraints (14, } x \text{ component), for } j = 1, \dots, P \\ \bullet \text{ kinematic constraints (15, } x \text{ component), for } j = 1, \dots, F \\ \bullet \text{ swing foot constraint (16)} \\ \bullet \text{ stability constraint (22).} \end{cases}$$

The first term of the cost function depends on all decision variables, directly on the ZMP and indirectly on the footsteps (through X_{mc}^k); its inclusion keeps the ZMP as much as possible close to the center of the moving constraint region. The second term minimizes ZMP variations in order to increase the smoothness of the resulting trajectory. The third term aims at realizing the candidate's footstep sequence as closely as possible.



An analogous problem QP- y is formulated for the y component, the only difference being that the kinematic constraint is given by the second expression in (15). Note that the ZMP constraints are obviously not enforced in time intervals belonging to flight phases; for the same reason, no ZMP decision variables are associated with such intervals. The constraints on the footstep placement are instead activated or deactivated depending on the gait type (walking/running) and the specific phase (single/double support, support/flight). **Figure 8** recapitulates this alternation.

The first ZMP sample (x_z^k, y_z^k) resulting from QP- x and QP- y is used in the prediction model to compute the next sample of the horizontal CoM trajectory (x_c^{k+1}, y_c^{k+1}) . Together with the result of QP- z , this gives the complete sample of the CoM trajectory. The first footstep position x_f^1, y_f^1 is instead used as target landing position for the swing foot in the next single support phase. CoM and swing foot trajectories are then tracked by the kinematic controller.

We conclude this section with a comment regarding the feasibility of QP- x and QP- y . Even if the stability constraint impacts the feasibility of the scheme (a detailed analysis for the planar walking case has been provided in Scianca et al. (2020)), there can be situations in which other constraints may conflict. For instance, reaching the assigned ground patch could conflict with the kinematic constraints if a strong deviation from the footstep plan has been previously required; in these situations, the horizontal QP can become unfeasible. This is a limitation that can be overcome by using a motion replanner rather than just relying on the local footstep adjustment capabilities of the controller.

6 A VELOCITY-DRIVEN FOOTSTEP PLANNER

This section describes a possible method for determining the required inputs for the proposed scheme in response to a high-level forward velocity command. We assume that the 3D map of the environment, called “elevation map”, is known. In addition to determining footstep placements and timing, we propose a simple criterion to decide for each step whether it will be realized by walking or running, and the corresponding reference CoM height relative to the ZMP.

The basic idea is that a change in velocity should be reflected in a change of both the spacing between the footsteps and their timing (Scianca et al., 2020). The goal is, given a desired velocity v , to determine a step length L and a step duration T such that $L/T = v$. The method assumes that a triplet of cruise parameters $(\bar{v}, \bar{L}, \bar{T})$ is available, where $\bar{v} = \bar{L}/\bar{T}$, with \bar{L} and \bar{T} denoting the typical step length and duration for the robot under consideration (for instance, the parameters of the default walking gait implemented by the robot manufacturer).

A deviation from \bar{v} should have an effect both on the step length and duration, which is expressed as:

$$v = \bar{v} + \Delta v = \frac{\bar{L} + \Delta L}{\bar{T} - \Delta T}.$$

By setting $\Delta L = \alpha \Delta T$ with $\alpha > 0$, one can derive an expression for the step duration realizing v

$$T = \bar{T} \frac{\alpha + \bar{v}}{\alpha + v},$$

which has to be divided into phases (single/double or support/flight). Such division can be directly chosen based on typical values for the ratio between phase durations in human walking and running. As a general rule based on human biomechanics (Novacheck, 1998), the double support duration should constitute the 20–40% of the step duration. Although this proportion should be inverted for flight phases during the run, here we will use for simplicity the same proportion in running gaits. The associated step length is easily calculated as $L = vT$.

This procedure outputs a larger L if a larger velocity v is commanded. It stands to reason that shorter steps should be realized by walking, whereas larger steps require running. In practice one may identify a step length threshold L_{\max} and command the scheme to generate a running step (mode = run) when $L > L_{\max}$, and a walking step (mode = walk) otherwise. Being this simple planner conceived for reference forward velocity tracking, the candidate’s footsteps on the y direction simply alternate with a fixed foot distance with respect to the initial CoM position.

So far, a triplet $(L^j, T^j, \text{mode}^j)$ has been generated for the j -th step in the preview horizon, from which it is immediate to compute the horizontal positions $(\hat{x}_f^j, \hat{y}_f^j)$ of the candidate footsteps. In

order to generate the vertical part of the plan, we need to retrieve from the elevation map the ground patch $\mathcal{P}_{\text{conv}}(\hat{x}_f^j, \hat{y}_f^j)$ corresponding to each footstep. If the footstep is unfeasible (e.g., because it is not completely contained in the patch), it is simply translated towards the previous footstep until feasibility is recovered. More sophisticated rules can be introduced, for example, enforcing a step with zero sagittal strides each time the two footsteps are located at two different heights⁷.

Finally, we need to assign the reference CoM height relative to the ZMP during the step, which will be used in (11). In particular, we choose a constant reference value $h^j = \bar{h}$ throughout the gait. This value can be modified if necessary, for example, to pass below an obstacle by crouching, as will be shown in the next result sections.

7 SIMULATIONS

In order to validate the proposed approach, we now present dynamic simulations on an HRP-4 humanoid robot in the DART dynamic environment. The actual torque limits of the platforms were not considered, as the primary objective here is to demonstrate the dynamic stability of a physically simulated robot performing dynamic gaits generated by our IS-MPC scheme. It should be noted in the fact that at the time of conducting this research, just a small percentage of the currently available humanoid robots would be able to perform dynamic running in a 3D environment, for example, Atlas by Boston Dynamics or ongoing projects such as the MIT humanoid (Chignoli et al., 2021) and Kangaroo by PAL Robotics⁸.

In all the simulations the kinematic controller runs at 100 Hz, while the vertical and horizontal QPs are solved using *hpipm* (Frison and Diehl, 2020). The simulations use the following parameters: sampling time $\delta = 0.01$ s, control horizon $T_c = 0.7$ s, preview horizon $T_p = 1.8$ s, $f_z^{\text{min}} = 114$ N, dimensions of the moving constraint region $d_{z,x} = d_{z,y} = 0.08$ m, dimensions of the kinematically admissible region $d_{a,x} = 1$ m, $d_{a,y} = 0.12$ m, with lateral displacement $\ell = 0.25$ m. The cost function weights of QP-z, QP-x and QP-y are, respectively, $\alpha_z = \beta_z = 10^{-5}$, $\alpha_x = \alpha_y = 1$, and $\beta_x = \beta_y = 10^4$.

Clips of all simulations are included in **Supplementary Video S1**.

⁷This prevents continuous stair climbing and descending, resulting in a less fluid but more robust gait. Such choice is quite common for experimental humanoid platforms (Caron et al., 2019b).

⁸A practical method to validate the result of our scheme on an actual robotic platform is to run a set of dynamic simulations. The generated motions can be evaluated based on the joint torques necessary to realize them, which can be directly obtained by querying the simulator. If the torques turn out to be excessive, one can try to modify some of the parameters in order to find a set that allows to produce a realizable motion (our results indicate that the duration of flight phases and step length are the most critical).

7.1 Walking on Flat Ground With Variable CoM Height

For this simulation, the ground is flat and therefore considered as a single horizontal patch. The candidate footstep positions are equally spaced by 0.15 m along x , and displaced by ± 0.18 m on y . Every step lasts 0.7 s, divided in 0.5 s for the single support and 0.2 s for the double support. The desired vertical CoM/ZMP displacement h varies throughout the simulation. It is initially 0.7 m for the first four steps, then is raised to 0.77 m for four subsequent steps. At each of the following five steps, it is progressively decreased by 0.02 m until reaching the final value of 0.58 m.

The simulation results are reported in **Figures 9, 10**. The stroboscopic motion of **Figure 9** also shows the actual CoM trajectory; note the tracking of the variable reference height. In the first plot of **Figure 10**, we display the CoM and ZMP trajectory in the (x, y) plane for the first eleven seconds of the gait. The CoM trajectory is clearly bounded with respect to the ZMP, proving the effectiveness of the proposed stability constraint. The second plot reports the vertical GRF computed by QP-z, which generates the variations of the CoM height. Note its smooth profile which is due to the last term of the cost function of QP-z. See the **Supplementary Video S1**.

7.2 Running on Flat Ground

For the running case, the spacing of the candidate footsteps along x is increased to 0.3 m, and the step duration is now 0.3 s for the single support phase and 0.15 s for the flight phase. At the start, an initial walking step is performed in preparation for running. The desired vertical CoM/ZMP displacement is kept constant at $h = 0.7$ m. The arm swing is enforced by a dedicated arm controller; its frequency is the same as the gait, while the amplitude is empirically assigned, similar to what was done in Kajita et al. (2002).

The results are shown in **Figures 11, 12**. The vertical CoM motion clearly exhibits the typical behavior observed in human biomechanics (Blickhan, 1989; Novacheck, 1998), consisting of the alternation between *absorption* and a *generation* phase: absorption is characterized by a horizontal deceleration and vertical descent of the CoM, while during the generation phase the CoM accelerates horizontally and rises in height until reaching an apex. **Figure 12** shows that the ZMP trajectory is discontinuous (consistent with the fact that the ZMP is not defined during the flight phases) and the CoM trajectory is bounded. The vertical GRF has a smooth profile while presenting small jumps when starting or concluding a support phase, due to constraints (9). However, the last term in the cost function of QP-z pushes the vertical GRF at the transition towards the minimum admissible value, which in itself is quite low with respect to the GRF peaks during running.

Overall, we were able to reach a maximum forward velocity of 1.25 m/s. In our opinion, this value is due to: *i*) the simplified model and *ii*) the kinematic controller. The effect of the simplified model could be alleviated by using a stabilizer based on the ZMP feedback. As for the kinematic controller, this can be improved by properly addressing the foot impact phase.

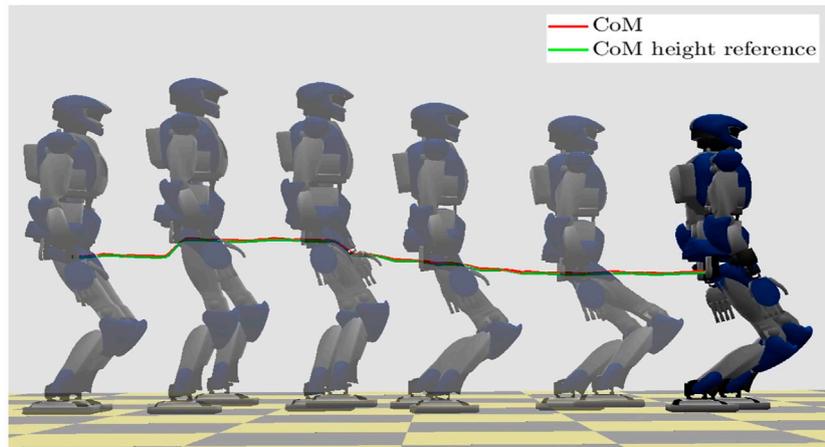


FIGURE 9 | Walking on flat ground with variable CoM height: stroboscopic motion.

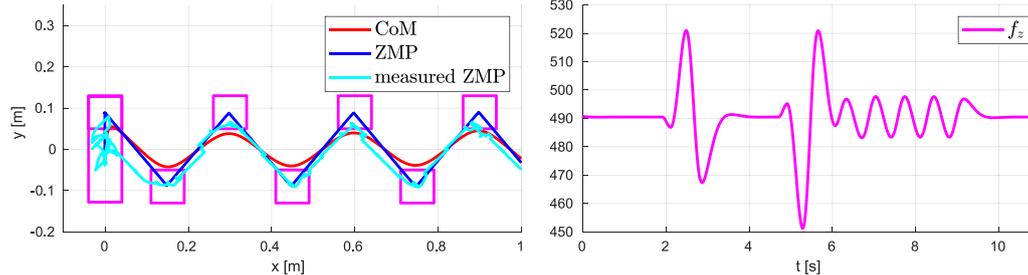


FIGURE 10 | Walking on flat ground with variable CoM height: CoM/ZMP trajectories (left) and vertical GRF (right).

The results are illustrated in more detail in **Supplementary Video S1**, in which we also included a push recovery simulation to further show the versatility of the scheme. In this simulation, HRP-4 is pushed twice by forces lasting 0.1 s of (50, 75, 0) N and (0, 75, 0) N, respectively. By adapting the footstep positions the robot can withstand the disturbances and proceed with the running gait. The same pushes would destabilize the robot if step adaptation was removed.

7.3 Walking and Running in a World of Stairs

In this simulation, the robot moves in a world of stairs. An additional overhanging obstacle was placed along the path, which requires the robot to lower its CoM in order to pass below it. The footstep plan is generated by the planner described in **Section 6**, where we have set $\alpha = 0.4$, $\bar{T} = 0.9$ s, $\bar{L} = 0.3$ m, and $L_{\max} = 0.35$ m. The reference velocity v is initially chosen as 0.3 m/s and then increased to 1.15 m/s after 16 s. The ground patches are assigned offline based on the elevation map as well as the reference CoM height, which is locally modified in order to let the robot pass below the overhanging obstacle.

Simulation results are reported in **Figures 13, 14**. The robot starts walking (mode = walk) with a step timing of

0.94 s (divided in 0.74 s for the single support phase and 0.2 s for the double support phase) and a step length along the sagittal direction of 0.28 m. When the reference velocity changes, the step length increases by 0.48 m, triggering a transition to running (mode = run); the corresponding durations are 0.24 s for the single support phase and 0.2 s for the flight phase. The sagittal velocity plot shows a satisfactory tracking of the reference sagittal velocity, while the CoM height follows the ground patch profile during both stair climbing and running.

See the **Supplementary Video S1**.

8 EXPERIMENT

In order to validate the proposed scheme on a physical platform, we tested the algorithm on the small humanoid OP3 by ROBOTIS. The same parameters adopted for the HRP-4 simulations were used, except for the sampling time $\delta = 0.025$ s (used both by the MPC and the kinematic controller), $f_z^{\min} = 5$ N, the dimensions of the moving constraint region, that are set to $d_{z,x} = d_{z,y} = 0.05$ m, as well as those of the kinematically

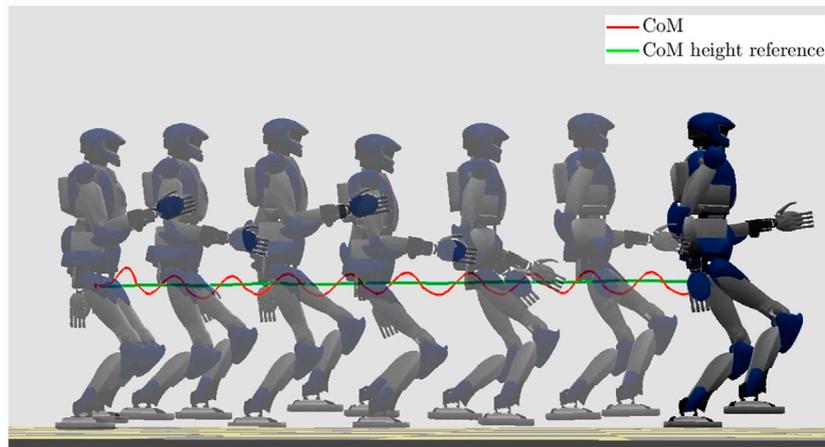


FIGURE 11 | Running on flat ground: stroboscopic motion.

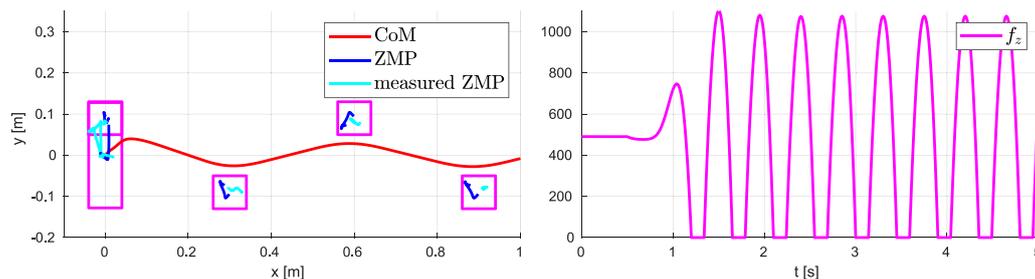


FIGURE 12 | Running on flat ground: CoM/ZMP trajectories (left) and vertical GRF (right).

admissible region, that have been changed to $d_{a,x} = 0.3$ m, $d_{a,y} = 0.05$ m, with lateral displacement $\ell = 0.1$ m.

The environment is composed of two ground patches, that is, the floor on which the robot is initially standing and slightly elevated ground. Along its path, the robot will encounter an overhanging obstacle, which has to be avoided by lowering the CoM. The footstep plan is generated with the following parameters: $\alpha = 0.4$, $\bar{T} = 0.6$ s, $\bar{L} = 0.07$ m, $L_{\max} = 0.15$ m. The reference velocity is chosen as 0.12 m/s. The candidate step length in the x direction is 0.07 m, while in the y direction the distance between the footsteps is 0.09 m. The reference CoM height is set to 0.22 m at the beginning of the experiment, lowered to overcome the obstacle, and then increased again.

The algorithm runs on the robot onboard computer (INTEL NUC with Intel Core i3-7100U 2×2.40 GHz processor and 8 GB 2133 MHz RAM) in real time. Each QP has a total of 28 decision variables ($T_c/\delta = 0.7/0.025$), while the number of constraints varies depending on which are active at a particular time (see **Figure 8**): QP- z always has exactly 28 constraints, while both QP- x and QP- y oscillate around 60.

Snapshots of the experiment are reported in **Figure 15**. This experiment validates the proposed scheme on a small-sized

humanoid, demonstrating its practical usability on a different platform with respect to the HRP-4 adopted for the dynamic simulations; see also the **Supplementary Video S1**.

9 RUNNING OVER TILTED SURFACES

The proposed method has been presented in the context of a world of stairs, where all contact surfaces are parallel. In this section, we show that it can be easily extended to the case of running over tilted surfaces. Although it would be possible to devise an extension for performing both walking and running motions, considering the case of running simplifies the problem. The reason is that, in a running gait, the robot is in contact with the tilted ground with a single foot at a time, and there is no double support phase. This is convenient because introducing double support over tilted surfaces would require adapting the definition of moving constraint to the case of nonparallel contact surfaces.

In **Section 5.2** we considered friction by means of a constraint on the vertical GRF (9). Avoiding slipping on tilted surfaces requires higher friction because the tangential component of the GRF must also compensate for the tangential component of

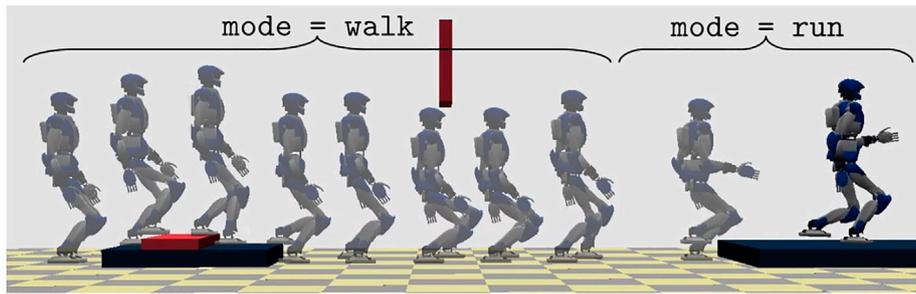


FIGURE 13 | Walking and running in a world of stairs: stroboscopic motion.

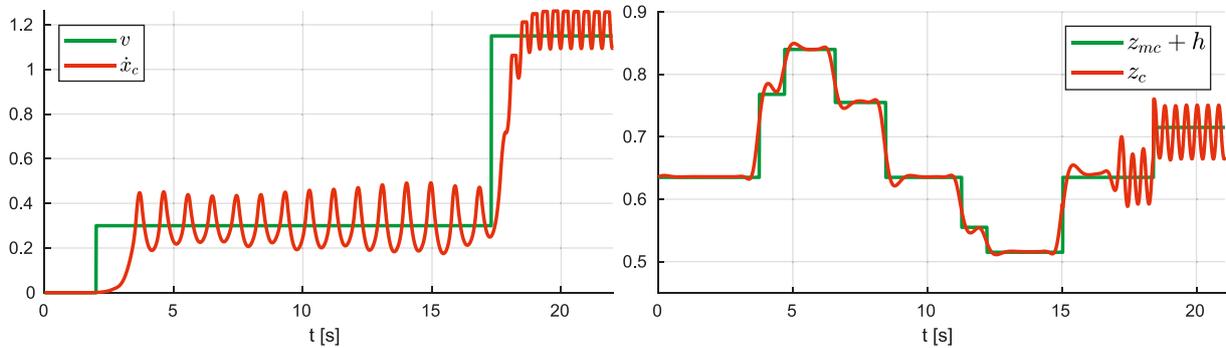


FIGURE 14 | Walking and running in a world of stairs: reference and actual sagittal velocity (left), reference and actual CoM height (right).

gravity. In order to account for this, it is sufficient to increase the minimum required vertical GRF by multiplying the right side of (9) by an appropriate coefficient which depends on the maximum slope of the ground. Since the environment is fully known, this can also be done on a per-footstep basis, by increasing the minimum GRF only when the foot is on a tilted surface, resulting in a less conservative GRF profile.

In order to maintain a formulation that is as close as possible to that presented in Section 5, we express the position of the CoM and ZMP in a frame that has the same orientation of the tilted patch (the z axis is the normal to the patch plane). In this frame, the vertical CoM dynamics assumes the expression:

$$\ddot{z}_c = \frac{f'_z}{m} - g'_z, \quad (25)$$

and the horizontal dynamics become:

$$\ddot{x}_c = \lambda(x_c - x_z) + g'_x, \quad (26)$$

$$\ddot{y}_c = \lambda(y_c - y_z) + g'_y, \quad (27)$$

where g'_x , g'_y and g'_z correspond to the component of the gravity vector in the tilted plane frame. This choice provides a model that is structurally equivalent to (25–27), with the addition of the horizontal gravity terms acting as known disturbances.

Below, we modify the gait generation scheme described in Sections 5.2 and 5.3 by writing the prediction model based on

(25–27) and performing an indirect compensation of the disturbance in the stability constraint (Smaldone et al., 2019).

In particular, the prediction model becomes:

$$\ddot{x}_c = \begin{cases} \lambda^{k+i}(x_c - x_z) + g'_x & \text{for } t \in [t_{k+i}, t_{k+i+1}) \quad i = 0, \dots, C-1 \\ \lambda_{LIP}(x_c - x_z) + g'_x & \text{for } t \geq t_{k+C}, \end{cases} \quad (28)$$

while the corresponding stability condition is given by the following result.

Proposition 3. Assume that $|x_z(t') - x_z(t)| \leq a + b(t' - t)$, $\forall t' \geq t$, for some $a, b > 0$, and that the current state (x_c^k, \dot{x}_c^k) satisfies

$$\begin{aligned} & G\Phi(t_{k+C}, t_k) \begin{pmatrix} x_c^k \\ \dot{x}_c^k \end{pmatrix} + \int_{t_k}^{t_{k+C}} G\Phi(t_{k+C}, \tau) B(\tau) x_z(\tau) d\tau \\ & = x_u^{k+C} - \int_{t_k}^{t_{k+C}} G\Phi(t_{k+C}, \tau) D g'_x(\tau) d\tau, \end{aligned} \quad (29)$$

where the disturbance matrix is $D = (0 \ 1)^T$. Then, system (28) is internally stable, that is, x_c is bounded with respect to x_z :

$$\exists M: |x_c(t) - x_z(t)| \leq M, \quad t \geq t_k.$$

Proof. Following the same steps of the proof of Prop. 2 for system (28) and using (Smaldone et al., 2019)

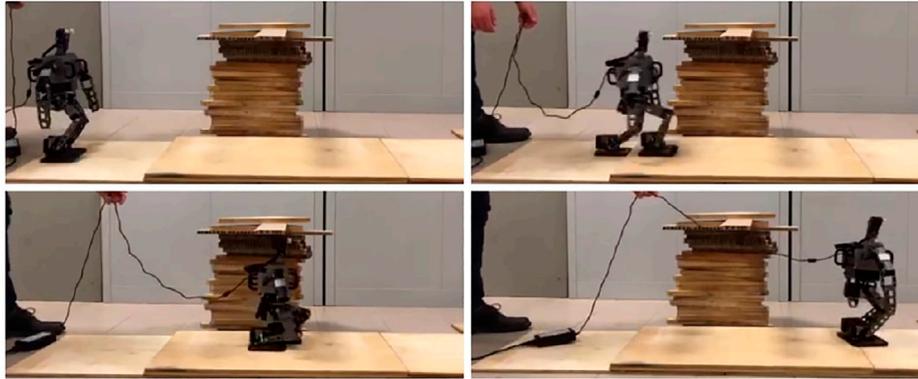


FIGURE 15 | Snapshots from the experiment. OP3 walks forward and steps onto an elevated ground patch then lower its CoM to crouch in order to pass below an obstacle. The cable attached to the robot's back is used for power supply only and does not sustain the robot while walking.

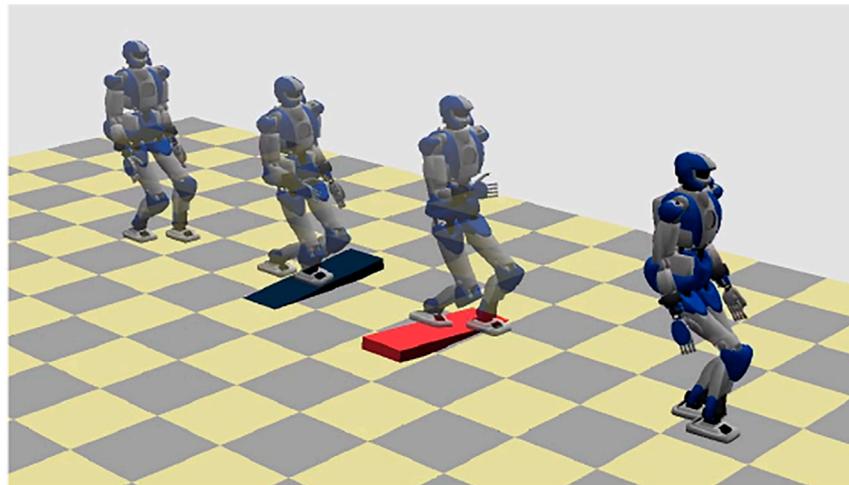


FIGURE 16 | Running over tilted surfaces: stroboscopic motion.

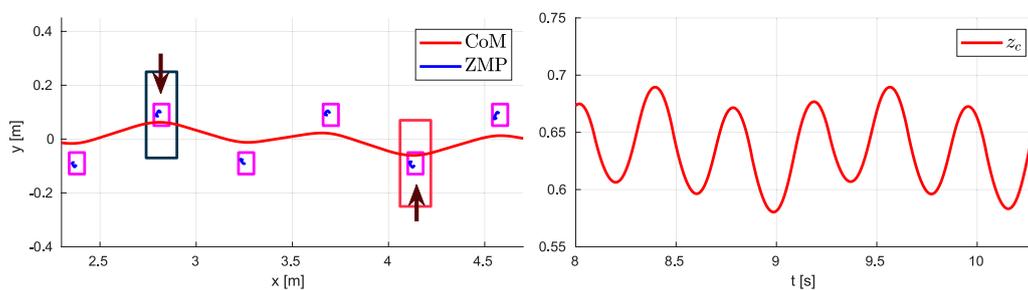


FIGURE 17 | Running over tilted surfaces. CoM/ZMP trajectories (left) and CoM height (right). The plots only refer to the portion of the simulation before and after the two tilted surfaces (represented as rectangles in the CoM/ZMP plots). The black arrow represents the equivalent disturbance caused by the tangential component of the gravity when stepping on the tilted patch.

$$x_u^k = \sqrt{\lambda_{LIP}} \int_{t_k}^{\infty} e^{-\sqrt{\lambda_{LIP}}(\tau-t_k)} x_z(\tau) d\tau - \frac{1}{\sqrt{\lambda_{LIP}}} \int_{t_k}^{\infty} e^{-\sqrt{\lambda_{LIP}}(\tau-t_k)} g_x'(\tau) d\tau,$$

we get (29).

Condition (29) can be written as a stability constraint as follows:

$$G \sum_{i=0}^{C-1} \prod_{j=i}^{C-1} \Phi_{k+j} B_{k+i} x_z^{k+i} = \sqrt{\lambda_{LIP}} \int_{t_{k+C}}^{\infty} e^{-\sqrt{\lambda_{LIP}}(\tau-t_{k+C})} \tilde{x}_z(\tau) d\tau - G \sum_{i=0}^{C-1} \prod_{j=i}^{C-1} \Phi_{k+j} D g_x'^{k+i} - G \prod_{i=0}^{C-1} \Phi_{k+i} \begin{pmatrix} x_c^k \\ \dot{x}_c^k \end{pmatrix}, \quad (30)$$

and used in the horizontal QP whenever required in order to step on tilted surfaces. Recall that the disturbance terms in (30) depend only on the tilting angle of the inclined planes and this information is assumed to be available from the footstep plan⁹.

9.1 Running Over Tilted Surfaces: Simulation

In this simulation the robot must run in an environment with two ramps (inclined by $\pm 8^\circ$) placed across the x path. A footstep plan is generated in response to the input velocity $v = 0.35$ m/s, which is switched to 1.1 m/s before the first ramp and then set back to the initial value after the second. We adopted the following parameters for the footstep planner: $\alpha = 0.4$, $\bar{T} = 0.7$ s, $\bar{L} = 0.3$ m, $L_{\max} = 0.35$ m.

Results are shown in **Figures 16, 17**. The CoM/ZMP trajectories, representing five steps of the running gait, highlight the typical effect of the proposed form of indirect disturbance compensation, consisting of a CoM trajectory that leans against the effect of the perturbation introduced by the tilted surface.

10 CONCLUSION

We presented a gait generation algorithm that extends our IS-MPC to both the case of 3D walking and running gaits. The proposed scheme is articulated as a two-stage MPC, and features a new stability constraint for the VH-IP model, ensuring bounded CoM trajectories with respect to the ZMP. A simple planner which provides the required input for the scheme is provided, as well as an extension to the case of running over tilted surfaces.

REFERENCES

Aboudonia, A., Scianca, N., De Simone, D., Lanari, L., and Oriolo, G. (2017). "Humanoid Gait Generation for Walk-To Locomotion Using Single-Stage

⁹This information is in practice also used to generate a swing foot trajectory that is consistent with the slope of the terrain.

Dynamic simulation on HRP-4 and an experiment on the OP3 humanoid are presented in order to validate the framework.

The strengths of the presented method can be summarized as:

- the effect of the CoM height is fully accounted for in the balance condition, without the need to bound height variations or approximating the horizontal dynamics;
- the resulting scheme is completely formulated as a set of quadratic optimizations subject to linear constraints, which allows for a very efficient MPC implementation, as demonstrated by it running onboard on the OP3 humanoid robot;
- the stability constraint included in the formulation provides a guarantee of bounded CoM/ZMP trajectories;
- the generated trajectories are very natural and compatible with what is observed in human walking/running, as reported by studies in biomechanics.

Future work will be aimed at extending the framework in order to include features to achieve robustness, as well as the integration of the scheme with a more sophisticated footstep planner. Furthermore, the time-varying formulation of the stability constraint can be abstracted from the specific setting of variable height gait generation, and possibly applied to a different scenario modeled as a time-varying unstable system.

DATA AVAILABILITY STATEMENT

The raw data supporting the conclusion of this article will be made available by the authors, without undue reservation.

AUTHOR CONTRIBUTIONS

All authors contributed to the conceptual development of this work. FS and NS implemented the proposed method and wrote the initial draft. All authors revised and approved the submitted version.

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: <https://www.frontiersin.org/articles/10.3389/frobt.2022.876613/full#supplementary-material>

- MPC," in 17th IEEE-RAS Int. Conf. on Humanoid Robots, 178–183. doi:10.1109/humanoids.2017.8239554
- Ahn, D., and Cho, B. (2021). Online Jumping Motion Generation via Model Predictive Control. *IEEE Trans. Industrial Electron.*, 4957–4965. doi:10.1109/TIE.2021.3078396
- Blickhan, R. (1989). The Spring-Mass Model for Running and Hopping. *J. Biomechanics* 22, 1217–1227. doi:10.1016/0021-9290(89)90224-8
- Brasseur, C., Sherikov, A., Collette, C., Dimitrov, D., and Wieber, P.-B. (2015). "A Robust Linear MPC Approach to Online Generation of 3D Biped Walking

- Motion," in 2015 IEEE-RAS Int. Conf. on Humanoid Robots, 595–601. doi:10.1109/humanoids.2015.7363423
- Caron, S., Escande, A., Lanari, L., and Mallein, B. (2019a). Capturability-based Pattern Generation for Walking with Variable Height. *IEEE Trans. Robotics* 36, 517–536. doi:10.1109/TRO.2019.2923971
- Caron, S., Kheddar, A., and Tempier, O. (2019b). "Stair Climbing Stabilization of the HRP-4 Humanoid Robot Using Whole-Body Admittance Control," in 2019 IEEE Int. Conf. on Robotics and Automation, 277–283. doi:10.1109/icra.2019.8794348
- Chignoli, M., Kim, D., Stanger-Jones, E., and Kim, S. (2021). "The MIT Humanoid Robot: Design, Motion Planning, and Control for Acrobatic Behaviors," in 2020 IEEE-RAS Int. Conf. on Humanoid Robots, 1–8. doi:10.1109/humanoids47582.2021.9555782
- Deits, R., and Tedrake, R. (2014). Footstep Planning on Uneven Terrain with Mixed-Integer Convex Optimization. 2014 IEEE-RAS Int. Conf. on Humanoid Robots, 279–286. doi:10.1109/humanoids.2014.7041373
- Deits, R., and Tedrake, R. (2015). "Computing Large Convex Regions of Obstacle-free Space through Semidefinite Programming," in Algorithmic Foundations of Robotics XI (Springer), 109–124. doi:10.1007/978-3-319-16595-0_7
- Engelsberger, J., Kozlowski, P., Ott, C., and Albu-Schäffer, A. (2016). Biologically Inspired Deadbeat Control for Running: from Human Analysis to Humanoid Control and Back. *IEEE Trans. Robot.* 32, 854–867. doi:10.1109/tro.2016.2581199
- Engelsberger, J., Ott, C., and Albu-Schäffer, A. (2013). "Three-dimensional Bipedal Walking Control Using Divergent Component of Motion," in 2013 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2600–2607. doi:10.1109/iros.2013.6696723
- Frison, G., and Diehl, M. (2020). *Hpipm: a High-Performance Quadratic Programming Framework for Model Predictive Control*. *IFAC-ArticlesOnLine* 53, 6563–6569. doi:10.1016/j.ifacol.2020.12.073
- Griffin, R. J., Wiedebach, G., Bertrand, S., Leonessa, A., and Pratt, J. (2017). "Walking Stabilization Using Step Timing and Location Adjustment on the Humanoid Robot, Atlas," in 2017 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 667–673. doi:10.1109/iros.2017.8202223
- Herd, A., Diedam, H., Wieber, P.-B., Dimitrov, D., Mombaur, K., and Diehl, M. (2010). Online Walking Motion Generation with Automatic Footstep Placement. *Adv. Robot.* 24, 719–737. doi:10.1163/016918610x493552
- Herd, A., Perrin, N., and Wieber, P.-B. (2012). "LMPC Based Online Generation of More Efficient Walking Motions," in 2012 IEEE-RAS Int. Conf. on Humanoid Robots, 390–395. doi:10.1109/humanoids.2012.6651549
- Hopkins, M. A., Hong, D. W., and Leonessa, A. (2014). "Humanoid Locomotion on Uneven Terrain Using the Time-Varying Divergent Component of Motion," in 2014 IEEE-RAS Int. Conf. on Humanoid Robots, 266–272. doi:10.1109/humanoids.2014.7041371
- Hu, K., Ott, C., and Lee, D. (2014). "Online Human Walking Imitation in Task and Joint Space Based on Quadratic Programming," in 2014 IEEE Int. Conf. on Robotics and Automation, 3458–3464. doi:10.1109/icra.2014.6907357
- Kajita, S., Kanehiro, F., Kaneko, K., Fujiwara, K., Harada, K., Yokoi, K., et al. (2003). "Biped Walking Pattern Generation by Using Preview Control of Zero-Moment Point," in 2003 IEEE Int. Conf. on Robotics and Automation, 1620–1626.
- Kajita, S., Nagasaki, T., Kaneko, K., and Hirukawa, H. (2007). ZMP-based Biped Running Control. *IEEE Robot. Autom. Mag.* 14, 63–72. doi:10.1109/mra.2007.380655
- Kajita, S., Nagasaki, T., Yokoi, K., Kaneko, K., and Tanie, K. (2002). "Running Pattern Generation for a Humanoid Robot," in 2002 IEEE Int. Conf. on Robotics and Automation, 2755–2761.
- Kamioka, T., Watabe, T., Kanazawa, M., Kaneko, H., and Yoshiike, T. (2015). "Dynamic Gait Transition Between Bipedal and Quadrupedal Locomotion," in 2015 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2195–2201. doi:10.1109/iros.2015.7353671
- Koolen, T., Posa, M., and Tedrake, R. (2016). "Balance Control Using Center of Mass Height Variation: Limitations Imposed by Unilateral Contact," in 2016 IEEE-RAS Int. Conf. on Humanoid Robots, 8–15. doi:10.1109/humanoids.2016.7803247
- Li, S., Chen, H., Zhang, W., and Wensing, P. M. (2021). "Quadruped Robot Hopping on Two Legs," in 2021 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 7448–7455. doi:10.1109/iros51168.2021.9636120
- Montejano, L. (1996). Some Results about Minkowski Addition and Difference. *Mathematika* 43, 265–273. doi:10.1112/s0025579300011761
- Novacheck, T. F. (1998). The Biomechanics of Running. *Gait Posture* 7, 77–95. doi:10.1016/s0966-6362(97)00038-6
- Popovic, M. B., Goswami, A., and Herr, H. (2005). Ground Reference Points in Legged Locomotion: Definitions, Biological Trajectories and Control Implications. *Int. J. robotics Res.* 24, 1013–1032. doi:10.1177/0278364905058363
- Pratt, J., Carff, J., Drakunov, S., and Goswami, A. (2006). "Capture Point: A Step toward Humanoid Push Recovery," in 6th IEEE-RAS Int. Conf. on Humanoid Robots, 200–207. doi:10.1109/ichr.2006.321385
- Raibert, M. H. (1986). *Legged Robots that Balance*. Cambridge, MS: MIT press.
- Sardain, P., and Bessonnet, G. (2004). Forces Acting on a Biped Robot. Center of Pressure-Zero Moment Point. *IEEE Trans. Syst. Man. Cybern. A* 34, 630–637. doi:10.1109/tsmca.2004.832811
- Scianca, N., Cognetti, M., De Simone, D., Lanari, L., and Oriolo, G. (2016). "Intrinsically Stable MPC for Humanoid Gait Generation," in 16th IEEE-RAS Int. Conf. on Humanoid Robots, 101–108. doi:10.1109/humanoids.2016.7803336
- Scianca, N., De Simone, D., Lanari, L., and Oriolo, G. (2020). MPC for Humanoid Gait Generation: Stability and Feasibility. *IEEE Trans. Robot.* 36, 1171–1188. doi:10.1109/tro.2019.2958483
- Scianca, N., Modugno, V., Lanari, L., and Oriolo, G. (2017). "Gait Generation via Intrinsically Stable MPC for a Multi-Mass Humanoid Model," in 2017 IEEE-RAS Int. Conf. on Humanoid Robots, 547–552. doi:10.1109/humanoids.2017.8246926
- Secer, G., and Cinar, A. L. (2020). "A Momentum-Based Foot Placement Strategy for Stable Postural Control of Robotic Spring-Mass Running with Point Feet," in 2020 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 3620–3626. doi:10.1109/iros45743.2020.9340980
- Secer, G., and Saranlı, U. (2018). Control of Planar Spring-Mass Running through Virtual Tuning of Radial Leg Damping. *IEEE Trans. Robot.* 34, 1370–1383. doi:10.1109/tro.2018.2830394
- Shafiee-Ashtiani, M., Yousefi-Koma, A., and Shariat-Panahi, M. (2017). "Robust Bipedal Locomotion Control Based on Model Predictive Control and Divergent Component of Motion," in 2017 IEEE Int. Conf. on Robotics and Automation, 3505–3510. doi:10.1109/icra.2017.7989401
- Smaldone, F. M., Scianca, N., Modugno, V., Lanari, L., and Oriolo, G. (2019). "Gait Generation Using Intrinsically Stable MPC in the Presence of Persistent Disturbances," in 19th IEEE-RAS Int. Conf. on Humanoid Robots, 651–656. doi:10.1109/humanoids43949.2019.9035068
- Sugihara, T., Imanishi, K., Yamamoto, T., and Caron, S. (2021). "3D Biped Locomotion Control Including Seamless Transition between Walking and Running via 3D ZMP Manipulation," in 2021 IEEE Int. Conf. on Robotics and Automation, 6258–6263. doi:10.1109/icra48506.2021.9561503
- Tajima, R., Honda, D., and Suga, K. (2009). "Fast Running Experiments Involving a Humanoid Robot," in 2009 IEEE Int. Conf. on Robotics and Automation, 1571–1576.
- Takenaka, T., Matsumoto, T., and Yoshiike, T. (2009a). "Real-time Motion Generation and Control for Biped Robot - 1st Report: Walking Gait Pattern Generation," in 2009 Int. Conf. on Intelligent Robots and Systems, 1084–1091.
- Takenaka, T., Matsumoto, T., Yoshiike, T., and Shirokura, S. (2009b). "Real-time Motion Generation and Control for Biped Robot - 2nd Report: Running Gait Pattern Generation," in 2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 1092–1099.
- Wensing, P. M., and Orin, D. E. (2013). "High-speed Humanoid Running through Control with a 3D-SLIP Model," in 2013 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 5134–5140. doi:10.1109/iros.2013.6697099
- Wieber, P. B. (2006). "Trajectory Free Linear Model Predictive Control for Stable Walking in the Presence of Strong Perturbations," in 2006 6th IEEE-RAS Int. Conf. on Humanoid Robots, 137–142. doi:10.1109/ichr.2006.321375
- Winkler, A. W., Bellicoso, C. D., Hutter, M., and Buchli, J. (2018). Gait and Trajectory Optimization for Legged Systems through Phase-Based End-Effector Parameterization. *IEEE Robot. Autom. Lett.* 3, 1560–1567. doi:10.1109/lra.2018.2798285
- Xiong, X., and Ames, A. D. (2018). "Bipedal Hopping: Reduced-Order Model Embedding via Optimization-Based Control," in 2018 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 3821–3828. doi:10.1109/iros.2018.8593547

Zamparelli, A., Scianca, N., Lanari, L., and Oriolo, G. (2018). Humanoid Gait Generation on Uneven Ground Using Intrinsically Stable MPC. *IFAC-ArticlesOnLine* 51, 393–398. doi:10.1016/j.ifacol.2018.11.574

Conflict of Interest: The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's Note: All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of

the publisher, the editors, and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

Copyright © 2022 Smaldone, Scianca, Lanari and Oriolo. This is an open-access article distributed under the terms of the Creative Commons Attribution License (CC BY). The use, distribution or reproduction in other forums is permitted, provided the original author(s) and the copyright owner(s) are credited and that the original publication in this journal is cited, in accordance with accepted academic practice. No use, distribution or reproduction is permitted which does not comply with these terms.