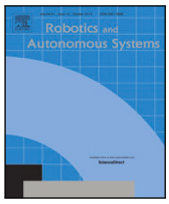
Robotics and Autonomous Systems 189 (2025) 104957

Contents lists available at ScienceDirect

Robotics and Autonomous Systems

journal homepage: www.elsevier.com/locate/robot



A feasibility-driven MPC scheme for robust gait generation in humanoids

Nicola Scianca [*,1], Filippo M. Smaldone, Leonardo Lanari, Giuseppe Oriolo

Dipartimento di Ingegneria Informatica, Automatica e Gestionale, Sapienza Università di Roma, via Ariosto 25, 00185 Roma, Italy

ARTICLE INFO

Keywords:
Humanoid
Robust locomotion
Model Predictive Control

ABSTRACT

We present a Robust Intrinsically Stable Model Predictive Control (RIS-MPC) framework for humanoid gait generation, which realizes as closely as possible a predefined sequence of footsteps in the presence of both persistent and impulsive perturbations. The MPC-based controller has two modes of operations, each involving a Quadratic Program. Since perturbations act by modifying the state, as well as the feasibility region itself, the fundamental idea is to select in real time the operation mode based on the feasibility properties of the current state. In *standard mode*, footsteps are regarded as fixed and the MPC computes a Center of Mass (CoM) and a Zero Moment Point (ZMP) trajectory. Robustness is ensured by a robust stability constraint which uses a disturbance estimate and by restricted ZMP constraints along the control horizon. In the presence of strong perturbations, that violate the aforementioned conditions, the system switches to *recovery mode*, in which footsteps positions and timings can be modified in order to recover feasibility. We analyze the feasibility of both modes of operation and provide conditions for recursive feasibility of the standard mode. Simulations on an HRP-4 robot as well as experiments on NAO and OP3 are provided to validate the scheme.

1. Introduction

Biped locomotion allows humans to move with great flexibility through unstructured environments. Realizing this kind of locomotion on a humanoid robot, however, is a challenging task, as it is necessary to satisfy the crucial requirement of maintaining balance at all times. This is further complicated by the fact that the humanoid dynamics are fundamentally unstable.

For controller design, different possible template models are available; the Linear Inverted Pendulum (LIP) model is the mostly adopted as it captures the essential unstable dynamic behavior of the humanoid. It describes the relation between the Center of Mass (CoM) and the Zero Moment Point (ZMP), the point w.r.t. which the horizontal moment of the contact forces is zero. Balance is achieved by keeping the ZMP inside the support polygon, i.e., the convex hull of the contact surfaces. The linearity of the LIP model allows to efficiently forecast the system evolution, and therefore enables approaches based on Model Predictive Control (MPC) [1]. This technique computes optimal inputs and trajectories over a prediction horizon, and allows enforcing hard constraints on state and inputs, both in equality and inequality form. Recent advances in numerical optimization methods and computing hardware make MPC algorithms extremely fast, and therefore amenable to real-time implementation.

A robust controller should satisfy the requirements of balance and stability also when the humanoid is subject to different types of perturbations. However, the complex nature of the problem implies that there is no single way to achieve such robustness, but rather a number of complementary approaches should be devised. For example, modifying footstep positions and timing for a humanoid is highly effective for absorbing impulsive pushes, but not as effective for counteracting persistent disturbances.

The problem of designing a robust MPC has attracted a great deal of research in general. Since MPC is based on solving a constrained optimization problem, disturbances can make the optimization problem infeasible. A possible approach uses the concept of positive invariant sets leading to the *tube-based* MPC [2–4]. The idea is that, for disturbances belonging to a compact convex polyhedron, it is possible to use the concept of robust positive invariance and constraint restriction in order to ensure that the MPC is able to satisfy the original constraints even under disturbances. However, direct application of tube-based MPC to humanoid gait generation is not straightforward, as most algorithms are designed for set-point regulation or state reference trajectory tracking, and require in general the system to be stabilized and the constraints to be time-invariant. An application of these ideas to bipeds can be found in [5], which uses robust positive invariance

* Corresponding author.
E-mail addresses: scianca@diag.uniroma1.it (N. Scianca), smaldone@diag.uniroma1.it (F.M. Smaldone), lanari@diag.uniroma1.it (L. Lanari), oriolo@diag.uniroma1.it (G. Oriolo).

[1] Nicola Scianca has been fully supported by PNRR MUR project PE0000013-FAIR.

https://doi.org/10.1016/j.robot.2025.104957
Received 15 December 2023; Received in revised form 21 December 2024; Accepted 26 February 2025
Available online 11 March 2025
0921-8890/© 2025 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY license (http://creativecommons.org/licenses/by/4.0/).

and a feedforward/feedback stabilizing control law. A interesting comparative study between tube-based and stochastic MPC is performed in [6].

The approaches just described require known disturbance bounds and can be very conservative. A more targeted approach consists in estimating the disturbance and performing some kind of compensation. In [7], several state estimators and disturbance observers based on the LIP dynamics are presented, while [8] shows a practical method to retrieve a measurement of an external force acting on the robot by using the inertial sensor and the force sensors located at the feet. Other studies [9,10] focus on the observation of external forces in the case of small-sized humanoids where force/torque sensors at the feet are not available. The measurement of the external force acting on the humanoid is used within a preview controller in [11], while other approaches [12,13] rely on the so-called divergent component of motion for the design of robust gait controllers based on the disturbance measurement. The MPC-based walking pattern generator of [14] exploits instead the measurement of the interaction force coming from the physical collaboration between a human and a humanoid, which is propagated along the prediction horizon in order to generate perturbation-aware CoM and ZMP trajectories.

The above approaches can generally deal with disturbances that are persistent and slowly-varying, but can fail if they change abruptly or are impulsive, thus becoming difficult to estimate. However, humanoid robots have the capability to absorb strong perturbations by adapting the position and timing of the footsteps. In MPC, real-time adaptation of the footstep positions can be realized efficiently using a linear formulation [15,16]. On the contrary, adapting the step timing can be much more complicated, as it usually involves nonlinear constraints or discrete optimization. The authors of [17] propose a method for adapting step positions and timings based on nonlinear optimization. A nonlinear MPC formulation is adopted in [18] where step timings are modified using time scaling. A two-stage MPC algorithm is presented in [19], where CoM and swing foot trajectories are interpolated and then timing is optimized. In [20], the authors propose a method for fast analytical computation of step positions and timings, focusing on walking over stepping stones. Other MPC schemes make use of Mixed-Integer Quadratic Programming (MIQP), where the nonlinearities introduced by step timing adaptation can be handled by adding discrete variables [21,22], at the expense of an increased computational load. Finally, when only the current step timing is adapted, an appropriate change of variables allows for a Quadratic Program (QP) formulation, such as in [23,24]. A similar idea is also used in [25], in which motion is sped up in the direction of the disturbance. Some of these works will be discussed in more detail in Section 11.7, where they will be compared with the method proposed in this paper.

All the techniques discussed so far address a specific robustness problem, by selectively focusing on either persistent or impulsive disturbances. Our objective is instead to design a unified MPC-based framework for gait generation which can handle generic disturbances; to this end, we will exploit the feasibility analysis of the underlying optimization problem.

Our work is based on the Intrinsically Stable MPC (IS-MPC) scheme initially proposed in [26], which includes an explicit constraint to guarantee internal stability, i.e., that the CoM trajectory is bounded with respect to the ZMP. A complete analysis of the scheme and its properties was carried out in [27] using the concept of *feasibility region*, the state space subset where the optimization problem is guaranteed to have a solution. In particular, it was shown that IS-MPC is *recursively feasible* (i.e., if it is feasible at a certain iteration it will remain such also in future iterations) and internally stable.

We successively proposed several extensions to IS-MPC in order to improve its robustness with respect to different types of disturbances. In [28] we considered the problem of persistent disturbances, such as the effect of a slope or of a weight attached to a robot limb; the proposed approach consisted in including a disturbance observer, and using its output to modify the stability constraint, achieving a form of indirect compensation. In [29], we investigated the restriction of the ZMP constraint as an additional tool for robustness; intuitively, restricting the ZMP constraint can be thought of as creating a safety margin which can be used to absorb perturbations. Although the idea was inspired by the tube-MPC literature, the restriction we introduced is conceptually different, as it changes along the MPC prediction horizon. Moreover, the ZMP restriction was designed in such a way to guarantee recursive feasibility against disturbances belonging to a known set. In [30], we also enabled step timing adaptation whenever a feasibility loss is imminent due, e.g., to an external disturbance. This is particularly effective for strong impulsive pushes, which cannot be readily detected by an observer and are too large to be conservatively included in a bounded set.

The framework we propose in this paper is a Robust Intrinsically Stable MPC (RIS-MPC) which extends and unifies the approaches just described using the concept of feasibility region as the key ingredient to achieve robustness. Two modes of operation are introduced, namely a *standard* and a *recovery* mode, whose key features are briefly summarized below.

- The standard mode puts together the disturbance observer, restricted ZMP constraints and a robust stability constraint in order to guarantee execution of the given footstep plan despite the presence of disturbances, as long as they belong to a certain expected set. In particular, a sufficient condition is derived for the corresponding QP to remain recursively feasible based on a feasibility analysis. The ZMP restriction function is optimally chosen in order to minimize the associated reduction of the feasibility region.
- The recovery mode is activated when the *feasibility check* reveals that the standard mode will become unfeasible at the next iteration, possibly due to an impulsive disturbance, or more in general to a disturbance outside the expected set. Accordingly, the ZMP restriction is disabled while automatic footstep placement with Step Timing Adaptation (STA) is enabled, in order to increase the feasibility region and recover feasibility if possible. After footsteps and timing are adapted, changes are made permanent by overwriting the original plan.

The proposed framework for robust gait generation is validated through dynamic simulations on the HRP-4 humanoid, as well as physical experiments on the small-sized robots NAO and OP3. The various scenarios were chosen so as to test the method under different types of disturbances, both persistent and impulsive, including terrain inclination, load transportation, lateral pushes, impacts with objects, and so on.

The paper is organized as follows. A general description of the proposed approach is given in Section 2. The standard mode and its feasibility properties are respectively presented in Section 3 and Section 4. A first set of LIP simulations are reported in Section 5 in order to illustrate the performance of the standard mode. The recovery mode is described in Section 6, while LIP simulations of the full RIS-MPC scheme are shown in Section 7. Section 8 briefly explains how the reference trajectories generated by the RIS-MPC are transformed to joint-level commands. Dynamic simulations on the HRP-4 robot are presented in Section 9, while experimental results on the two small-scale humanoids are reported in Section 10. Section 11 provides an exhaustive discussion on the role of the main assumptions and components of the method (including the expected disturbance set and the observer), some guidelines on the choice of the controller parameters, and a comparison (both qualitative and quantitative) with the state of the art. Possible future work is mentioned in the concluding section.

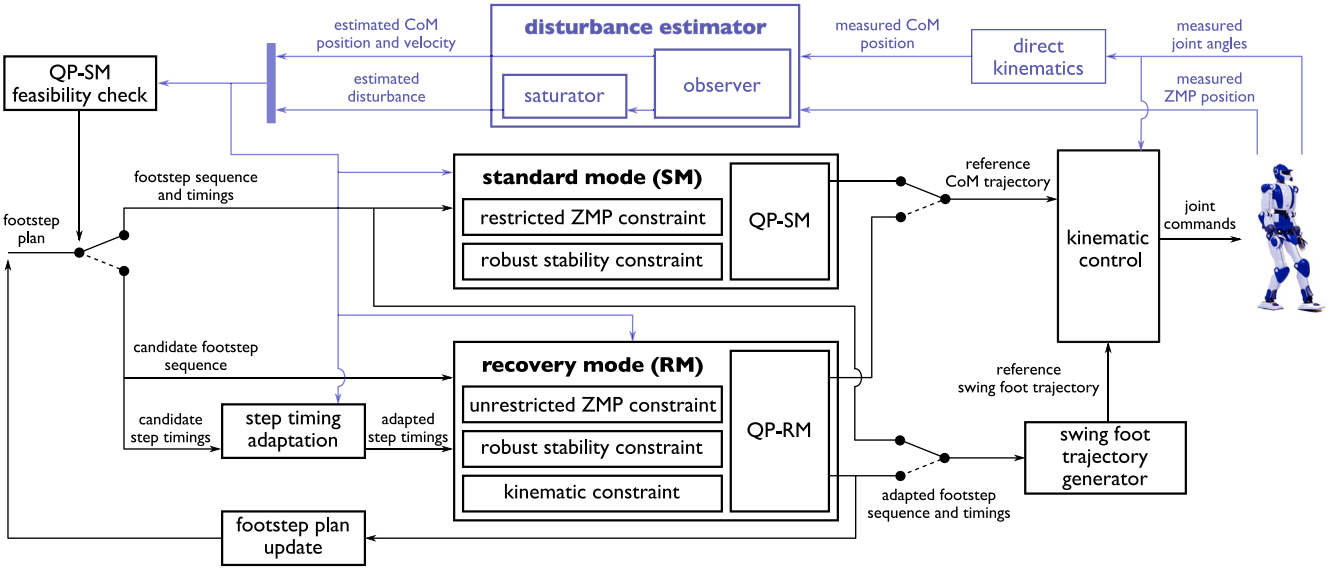


Fig. 1. The proposed Robust Intrinsically Stable MPC (RIS-MPC) control scheme. The switch elements are *up* in standard mode and *down* in recovery mode. The blue section of the scheme identifies parts related to sensing and estimation. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

2. The proposed approach

In this paper, we consider the problem of generating a stable humanoid gait that realizes as closely as possible a predefined sequence of footsteps in the presence of perturbations. These can be of different nature: they can be persistent, such as gravity when walking on an incline, and/or impulsive, such as a push or an impact with an obstacle.

At the core of the proposed framework is an MPC-based controller, which has two modes of operation, i.e., the *standard mode* (SM) and the *recovery mode* (RM). In the two modes, the controller solves different Quadratic Programs (QPs) over the control horizon, respectively called QP-SM and QP-RM; each QP is guaranteed to be solvable provided that the state of the robot belongs to the corresponding feasibility region. Since perturbations act by modifying the state, as well as the feasibility region itself, the fundamental idea is to select in real time the operation mode based on the feasibility properties of the current state.

A block scheme of the proposed solution approach, called Robust Intrinsically Stable MPC (RIS-MPC for brevity) is shown in Fig. 1. At each time instant t_k, a *footstep plan* (i.e., a sequence of footstep positions and associated timings) is received[2] as input. The algorithm performs a *feasibility check*, which verifies whether the robot state belongs to the feasibility region $\mathcal{F}_{\mathrm{S}}^k$ of QP-SM. In this case, the system can operate in standard mode, otherwise it must switch to recovery mode.

The standard mode (Section 3) is based on the Intrinsically Stable MPC (IS-MPC) scheme [27], which hinges upon a special constraint and guarantees stable gait generation in the absence of perturbations. In this paper, IS-MPC is made robust to perturbations by using specific strategies, which consist in *(i)* a robust version of the above stability constraint based on a disturbance estimate produced by a specific module, and *(ii)* a restriction of the ZMP constraint along the control horizon. It will be shown (Section 4) that recursive feasibility of the standard mode is ensured if the perturbations are bounded in a known set.

When QP-SM becomes unfeasible (i.e., when recursive feasibility is lost due to a perturbation outside the given bounds) the recovery mode (Section 6) is activated. Accordingly, QP-RM is modified with respect to QP-SM so as to enlarge the associated feasibility region $\mathcal{F}_{\mathrm{R}}^k$ as much as possible. To this end, footstep positions are included among the decision variables of QP-RM, so that they can be modified to maintain balance. Moreover, while the robust stability constraint of QP-SM is maintained, no restriction is performed on the ZMP constraint. In fact, the role of the ZMP restriction is to guarantee recursive feasibility of QP-SM under bounded disturbances in a known set: when feasibility is lost due to disturbances outside this set, maintaining the restriction would go against the recovery goal.

To increase the chance of success in recovery mode, QP-RM is preceded by a Step Timing Adaptation (STA) module. This is in charge of modifying the duration of the current step in such a way that the feasibility region $\mathcal{F}_{\mathrm{R}}^k$ of QP-RM for the chosen duration contains the robot state. STA is performed outside of QP-RM to maintain linearity of the constraints in the latter.

After solving the QP associated to the currently active mode, the reference CoM trajectory is available over the control horizon. The next footstep position and corresponding timestamp are either those specified by the original footstep plan (in standard mode) or those adapted by QP-RM (in recovery mode). These data are used to generate an appropriate reference trajectory for the swinging foot. Both references trajectories are finally sent to a kinematic controller which guarantees their execution by the robot.

3. Standard mode

The proposed RIS-MPC scheme operates over sampling intervals of duration δ, with the generic time instant denoted by $t_k = k\,\delta$. A k superscript indicates that the variable is sampled at t_k, e.g., $x(t_k) = x^k$. RIS-MPC works by solving a constrained optimization problem over a *control horizon* T_c, with $T_c = C\,\delta$.

At each t_k, the robot receives a footstep plan which extends over a *preview horizon* $T_p > T_c$ and consists of a sequence of footstep positions in the x-y plane with an associated timing (see Fig. 2). It is assumed that all footsteps have the same orientation with respect to the x axis, chosen to be zero without loss of generality. This simplifies the exposition as it decouples the problem along the x (sagittal) and y (coronal) axes, allowing us to present equations for the x axis only. Equations for y will be provided only when different from their x counterpart. The proposed method can however be extended in principle to the case of footsteps with variable orientation [27].

[2] These data typically come from a footstep planner, which is not discussed in this paper. See, e.g. [31,32] for flat ground planners and [33,34] for a 3D planner.

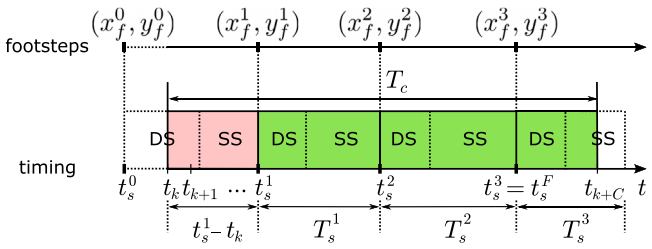


Fig. 2. Footsteps with timing within the MPC control horizon T_c. The current time is t_k. The remaining part of the current step is shown in red, whereas future steps inside the control horizon are shown in green. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

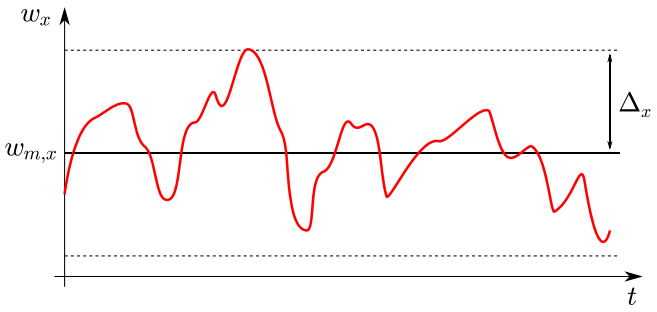


Fig. 3. An example of disturbance w_x with the associated expected disturbance set $\mathcal{W}_x$, defined by the midrange $w_{m,x}$ and the maximum deviation Δ_x.

The portion of the footstep plan contained in the control horizon T_c is denoted as

$$X_f^k = (x_f^1, \dots, x_f^F), \quad Y_f^k = (y_f^1, \dots, y_f^F), \quad \tau_s^k = (t_s^1, \dots, t_s^F).$$

Here, (x_f^j, y_f^j), $j = 1, \dots, F$, is the position of the jth footstep and t_s^j is the corresponding timestamp, indicating the time instant at which the (swinging) foot should land on the footstep, i.e., the beginning of double support (see Fig. 2). The duration of the jth step is given by $T_s^j = t_s^{j+1} - t_s^j$, divided (with fixed ratio) in a double support (DS) and a single support (SS) phase, with duration T_{ds}^j and T_{ss}^j, respectively.

In this section, we discuss in detail the prediction model and the various constraints that characterize the QP in standard mode. In doing so, we will introduce a model for the disturbance and describe the disturbance estimator.

3.1. Prediction model and expected disturbance set

The *Linear Inverted Pendulum* (LIP) is a template model of the humanoid dynamics that has proved quite effective in practice [35]. The equations of the LIP can be derived from the balance of moments around the ZMP by assuming a flat horizontal ground, a constant height $\bar{z}_c$ of the CoM, and constant centroidal angular momentum. One obtains a linear dynamics between the CoM coordinates (x_c, y_c) and the ZMP coordinates (x_z, y_z), which takes the following decoupled form

$$\ddot{x}_c = \eta^2(x_c - x_z) \tag{1}$$
$$\ddot{y}_c = \eta^2(y_c - y_z), \tag{2}$$

where $\eta = \sqrt{g/\bar{z}_c}$ is the pendulum natural frequency, with g the gravity acceleration.

Consider now an additive disturbance (w_x, w_y) acting on the LIP dynamics. The source of such disturbance can be external (e.g., a force) and/or internal (e.g., unmodeled dynamics). It is assumed that (w_x, w_y) belongs to an *expected disturbance set* $\mathcal{W} = \mathcal{W}_x \times \mathcal{W}_y$, where

$$\mathcal{W}_x = \{w_x : w_{m,x} - \Delta_x \le w_x \le w_{m,x} + \Delta_x\}, \tag{3}$$

and a similar expression holds for $\mathcal{W}_y$. In Eq. (3), $w_{m,x}$ is the midrange, i.e., the average between the upper and the lower bound of $\mathcal{W}_x$, and Δ_x is the *maximum deviation* from the midrange (see Fig. 3). In practice, these values can often be derived by analyzing the sources of disturbances that the robot is subject to.[3] As an alternative, one may perform a preliminary experimental motion campaign using any available controller for the considered humanoid, and assess the maximum disturbance by means of a suitable estimator, either off-line or on-line; for example, one may adopt the disturbance observer presented in Section 3.2. Clearly, the experimental conditions should bring into play the relevant disturbance sources, such as unmodeled dynamics and external perturbations (pushes, etc.). As a baseline controller, one may even use RIS-MPC (Section 3.5) with a commonsense initial guess on the disturbance bounds. See the concluding section for additional related comments.

Since the x- and y-dynamics in (1)–(2) are decoupled and formally identical, we focus hereafter on the perturbed dynamics along the x axis:

$$\ddot{x}_c = \eta^2(x_c - x_z) + w_x, \tag{4}$$

and refer to this model as the *perturbed* x-LIP.

Our MPC prediction model is the perturbed x-LIP with a dynamical extension on the input channel:

$$\begin{pmatrix} \dot{x}_c \\ \ddot{x}_c \\ \dot{x}_z \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ \eta^2 & 0 & -\eta^2 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_c \\ \dot{x}_c \\ x_z \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \dot{x}_z + \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} w_x \tag{5}$$

with the ZMP velocity $\dot{x}_z$ now acting as input. In particular, we assume piecewise-constant inputs (and thus piecewise-linear ZMP), i.e., $\dot{x}_z(t) = \dot{x}_z^k$ for $t \in [t_k, t_{k+1})$.

Note that the prediction model includes the disturbance w_x, whose value is typically unknown. As will be made clear in the following, any computational use of this model will in practice require that an estimate of w_x is available. Motivated by this, we now discuss the structure of a disturbance estimator.

3.2. Disturbance estimator

The proposed disturbance estimator consists of two consecutive stages (see Fig. 1). The first is a classical asymptotic observer, whose estimate is then processed by the second stage to make sure that certain properties (which will be instrumental in achieving robustness) are satisfied.

The inputs for the first stage are measurements of the positions of the CoM and the ZMP. The first can be reconstructed from joint encoder readings through direct kinematics, while the second requires either ankle force/torque sensors at the ankle [36] or force sensing resistors at the foot [37]. Note that the CoM and ZMP position may be either expressed in a global (inertial) frame or in a local frame, e.g., attached to the support foot. The estimate of the disturbance will be the same, as its computation only relies on the relative CoM-ZMP displacement. This also means that occasional slippage of the support foot would not have an effect on the ability of the observer to estimate a disturbance.

As a disturbance model, we adopt the following simple dynamics

$$\dot{w}_x = 0, \tag{6}$$

[3] For example, consider a humanoid walking on a slope with unknown inclination θ. The tangential component of gravity introduces a disturbance $w_x = g \sin\theta$. If the value of θ belongs to $[\theta_{\min}, \theta_{\max}]$, the corresponding expected disturbance set $\mathcal{W}_x$ is identified by

$$w_{m,x} = \frac{g}{2}(\sin\theta_{\max} + \sin\theta_{\min})$$
$$\Delta_x = \frac{g}{2}(\sin\theta_{\max} - \sin\theta_{\min}).$$

which, under state (velocity) resets, can produce a piecewise-linear disturbance. This is consistent with the assumption taken in the prediction model (5) that the ZMP position (which acts at the same level of the disturbance in the perturbed LIP) is itself piecewise-linear.

For the 5-dimensional extended system consisting of (5) and (6), we may design a reduced-order observer [38] which uses the available measurements, collected in a vector $v = (x_c, x_z)$, to reconstruct an estimate of the remaining states, collected in a vector $\chi = (\dot{x}_c, w_x, \dot{w}_x)$. Working out the computations for the extended system, we obtain the following structure for the observer:

$$\dot{\xi} = A\,\xi + L\,\dot{x}_z + M\,v \tag{7}$$

$$\hat{\chi} = \xi + Gv, \tag{8}$$

where

$$A = \begin{pmatrix} -g_{11} & 1 & 0 \\ -g_{21} & 0 & 1 \\ -g_{31} & 0 & 0 \end{pmatrix}, \quad L = \begin{pmatrix} -g_{12} \\ -g_{22} \\ -g_{32} \end{pmatrix},$$

$$M = \begin{pmatrix} \eta^2 - g_{11}^2 + g_{21} & -\eta^2 - g_{11}\,g_{12} + g_{22} \\ g_{31} - g_{11}\,g_{21} & g_{32} - g_{12}\,g_{21} \\ -g_{11}\,g_{31} & -g_{12}\,g_{31} \end{pmatrix},$$

and the g_{ij}'s are the elements of the 3×2 gain matrix G. In particular, the first column elements g_{i1} $(i = 1, 2, 3)$ must be chosen so as to make A Hurwitz (so that the dynamics of the observation error is asymptotically stable), while the second column elements g_{i2} $(i = 1, 2, 3)$ define[4] matrix L.

The second stage of the disturbance estimation module takes the estimate $\hat{w}_x^k$ (which is actually the second component of $\hat{\chi}$ sampled at t_k) generated by the reduced observer, and produces a modified estimate $\tilde{w}_x^k$ by solving the following QP, referred to as the *saturator*:

$$\begin{cases} \min_{\tilde{w}_x^k} (\tilde{w}_x^k - \hat{w}_x^k)^2 \\ \text{subject to} : \\ \bullet\ \text{bound on } \tilde{w}_x^k \\ \qquad w_{m,x} - \Delta_x \le \tilde{w}_x^k \le w_{m,x} + \Delta_x, \quad \text{(a)} \\ \bullet\ \text{bound on the variation of } \tilde{w}_x^k \\ \qquad |\tilde{w}_x^k - \tilde{w}_x^{k-1}| \le w_x^{\text{diff}}. \quad \text{(b)} \end{cases} \tag{9}$$

The first condition enforced by the saturator is that the final disturbance estimate $\tilde{w}_x$ must belong to the expected disturbance set $\mathcal{W}_x$, see (3). The second condition entails that the increment between consecutive estimates is bounded by a given quantity w_x^{diff} (clearly, it must be $w_x^{\text{diff}} \le 2\Delta_x$). These conditions will be needed later to prove that our RIS-MPC scheme, which makes use of the disturbance estimate $\tilde{w}_x$, is recursively feasible in the presence of disturbances (see Section 4.2).

3.3. Restricted ZMP constraint

Dynamic balance is ensured by keeping the ZMP at all times inside the support polygon, which is the convex hull of the contact areas. In the single support phase, enforcing this condition leads to a linear constraint, because the convex hull coincides with the (rectangular) footprint. In the double support phase, however, considering the full extension of the support polygon would result in a nonlinear constraint. For this reason, we use a *moving* constraint: i.e., we define an admissible ZMP region which has the same shape of the footprint and slides from one footstep to the other during this phase.

[4] Although the convergence of the observer for system (5)–(6) is guaranteed regardless of the choice of L, we have observed that in the application to the full humanoid model it is beneficial to use a nonzero L.

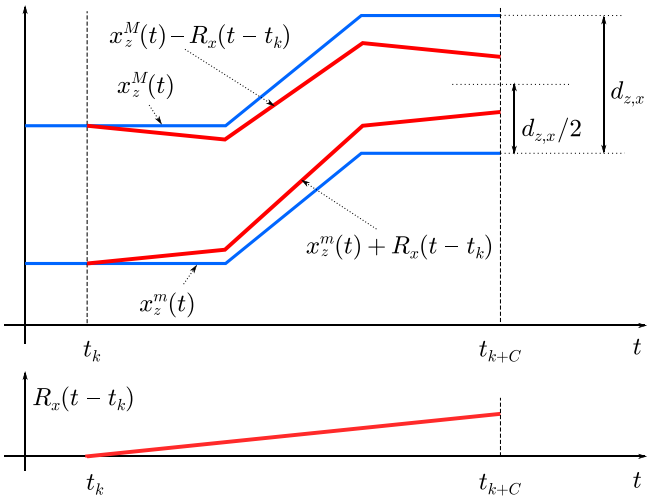


Fig. 4. The restricted ZMP constraint (above) generated by a linear restriction function (below).

In particular, assume that the footprint is a rectangle with dimensions $d_{z,x}, d_{z,y}$. The ZMP constraint within the control horizon can then be simply expressed as

$$x_z^m(t) \le x_z(t) \le x_z^M(t), \quad t \in [t_k, t_{k+C}], \tag{10}$$

where the bounds $x_z^m(t)$ and $x_z^M(t)$ satisfy

$$x_z^M(t) - x_z^m(t) = d_{z,x}, \quad \text{for all } t,$$

and must be computed on the basis of the footstep plan. More specifically, define the piecewise-linear function

$$\sigma(t, t_i, t_f) = \frac{1}{t_f - t_i}\left(r(t - t_i) - r(t - t_f)\right),$$

where $r(t)$ is the unit ramp starting at 0. The bounds of the ZMP constraint (10) during the jth step $(j = 0, \dots, F)$ can then be expressed as

$$x_z^m(t) = x_{\text{mc}}(t) - \frac{1}{2}\,d_{z,x} \qquad x_z^M(t) = x_{\text{mc}}(t) + \frac{1}{2}\,d_{z,x},$$

for $t \in [t_s^j, t_s^{j+1}0)$, having defined the center of the moving constraint as

$$x_{\text{mc}}(t) = x_f^j + (x_f^{j+1} - x_f^j)\sigma(t, t_s^j, t_s^j + T_{ds}^j).$$

In order to robustify the basic IS-MPC control scheme, we are going to enforce a modified version of the ZMP constraint (10) defined as

$$x_z^m(t) + R_x(t - t_k) \le x_z(t) \le x_z^M(t) - R_x(t - t_k), \quad t \in [t_k, t_{k+C}], \tag{11}$$

where the *restriction function* $R_x(t)$ must satisfy

$$0 \le R_x(t) \le d_{z,x}/2, \tag{12}$$

for $t \in [0, T_c]$. The left-hand side of this inequality guarantees that $R_x(t)$ never enlarges the admissible region, whereas the right-hand side implies that such region cannot vanish before the end of the control horizon. An illustrative example is shown in Fig. 4, with reference to a case where the ZMP constraint is restricted using a linear $R_x(t)$. In the following, we refer to (11) as the *restricted ZMP constraint*.

On the one hand, it is intuitive that enforcing a restricted constraint on the ZMP is beneficial for robustness, because it provides a safety margin for absorbing the effect of disturbances. On the other hand, it will also affect negatively the feasibility of the QP problem at t_k, because it is more difficult to satisfy the restricted constraint than its original version. In the following, we will analyze in detail both these issues, respectively in Sections 4.2 and 4.3. In particular, in the latter we will provide guidelines on how to choose the actual profile of the restriction function so as to minimize the impact on feasibility.

3.4. Robust stability constraint

With reference to the perturbed x-LIP model (4), define the following variable

$$x_u = x_c + \dot{x}_c/\eta, \tag{13}$$

known as *capture point* [39] or *divergent component of motion* [40]. The dynamics of x_u

$$\dot{x}_u = \eta\, x_u - \eta\, x_z + w_x/\eta \tag{14}$$

is clearly unstable due to the presence of a positive eigenvalue. As a consequence, gait generation is inherently unstable, as the CoM will in general diverge (exponentially) with respect to the ZMP, even if the latter is kept in the support polygon at all times. This results in a motion that would be balanced in theory but unrealizable in practice due to kinematic limitations. To overcome this issue, we impose a constraint aimed at avoiding divergence of the CoM, based on the following result.

Proposition 1. *Assume that the perturbed x-LIP (4) is initialized at*

$$x_u^k = \eta \int_{t_k}^{\infty} e^{-\eta(\tau - t_k)} x_z(\tau) d\tau - \frac{1}{\eta} \int_{t_k}^{\infty} e^{-\eta(\tau - t_k)} w_x(\tau) d\tau. \tag{15}$$

Then, the CoM position x_c remains bounded at all times with respect to the ZMP position x_z.

Proof. See Appendix. ■

We refer to (15) as the *stability condition* and rearrange it as

$$\eta \int_{t_k}^{t_{k+C}} e^{-\eta(\tau - t_k)} x_z(\tau) d\tau = x_u^k - c_x^k + c_{w,x}^k, \tag{16}$$

where

$$c_x^k = \eta \int_{t_{k+C}}^{\infty} e^{-\eta(\tau - t_k)} x_z(\tau) d\tau \tag{17}$$

$$c_{w,x}^k = \frac{1}{\eta} \int_{t_k}^{\infty} e^{-\eta(\tau - t_k)} w_x(\tau) d\tau. \tag{18}$$

Condition (16) is *non-causal*, as c_x^k and $c_{w,x}^k$ depend on the future values of x_z and w_x through (17) and (18), respectively. To obtain a causal version of the condition, we will address these two sources of non-causality separately.

The term c_x^k depends on the future trajectory of x_z beyond the control horizon. In [27], we proposed to replace $x_z(\tau)$ in (17) with a trajectory $\tilde{x}_z(\tau)$ conjectured on the basis of short-term information from the footstep plan over the preview horizon T_p. In particular, we define the *anticipative tail* as

$$\tilde{x}_z(t) = \begin{cases} x_{\mathrm{mc}}(t) & \text{for } t \in [t_{k+C}, t_{k+P}] \\ \tilde{x}_z(t_{k+P}) & \text{for } t \in (t_{k+P}, \infty), \end{cases} \tag{19}$$

which amounts to taking $\tilde{x}_z$ as the midpoint between the ZMP bounds $x_z^m(t)$ and $x_z^M(t)$ until the end of the preview horizon, and then maintaining the final value $\tilde{x}_z(t_{k+P})$ indefinitely. Using the anticipative tail, we define a computable version of c_x^k as

$$\tilde{c}_x^k = \eta \int_{t_{k+C}}^{\infty} e^{-\eta(\tau - t_k)} \tilde{x}_z(\tau) d\tau. \tag{20}$$

The second source of non-causality in (16) is $c_{w,x}^k$, which depends on the evolution of the disturbance w_x after t_k. To obtain a computable version of this term, we make use of the estimate $\tilde{w}_x$ provided by the disturbance estimator of Section 3.2. In particular, since the estimator does not perform any kind of prediction, we simply let

$$w_x(t) = \tilde{w}_x^k, \quad t \in [t_k, \infty), \tag{21}$$

i.e., we use the current estimate $\tilde{w}_x^k$ throughout the integration interval. Thus, we obtain

$$\tilde{c}_{w,x}^k = \frac{1}{\eta} \int_{t_k}^{\infty} e^{-\eta(\tau - t_k)} \tilde{w}_x^k d\tau = \frac{\tilde{w}_x^k}{\eta^2}. \tag{22}$$

Wrapping up, the causal version of the stability condition (16) is written as

$$\eta \int_{t_k}^{t_{k+C}} e^{-\eta(\tau - t_k)} x_z(\tau) d\tau = x_u^k - \tilde{c}_x^k + \tilde{c}_{w,x}^k, \tag{23}$$

with $\tilde{c}_x^k$ and $\tilde{c}_{w,x}^k$ given by Eqs. (20) and (22), respectively.

Note that the left-hand side of (23) depends on the MPC decision variables in standard mode, i.e., the ZMP velocity inputs $\dot{x}_z^k, \dots, \dot{x}_z^{k+C-1}$ within the control horizon. To make this dependence explicit, we can write x_z as

$$x_z(t) = x_z^k + \sum_{i=0}^{C-1} \left(r(t - t_{k+i}) - r(t - t_{k+i+1}) \right) \dot{x}_z^{k+i},$$

and then plug this in (23), obtaining after integration the following causal constraint on the velocity inputs:

$$\sum_{i=0}^{C-1} (e^{-i\eta\delta} - e^{-(i+1)\eta\delta}) \frac{\dot{x}_z^{k+i}}{\eta} = x_u^k - x_z^k (1 - e^{-\eta T_c}) - \tilde{c}_x^k + \tilde{c}_{w,x}^k. \tag{24}$$

In the following we refer to (24) as the *robust stability constraint*. In [27], it was shown that the stability constraint for the nominal case (that is, for $w_x = 0$) can be rewritten as an equivalent constraint on the state at the end of the horizon, i.e., a *terminal* constraint; it is trivial to verify that the same can be done for the robust version (24).

3.5. Standard mode QP

The proposed RIS-MPC scheme computes the ZMP control action by considering motion of the LIP both in the x and y directions. For the latter, similarly to (5), we define as prediction model a perturbed y-LIP with dynamical extension:

$$\begin{pmatrix} \dot{y}_c \\ \ddot{y}_c \\ \dot{y}_z \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ \eta^2 & 0 & -\eta^2 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} y_c \\ \dot{y}_c \\ y_z \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \dot{y}_z + \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix} w_y, \tag{25}$$

where w_y is in practice replaced by an estimate $\tilde{w}_y$, computed by a disturbance estimator analogous to the one discussed in Section 3.2. For the motion in y, the restricted ZMP constraint is expressed as

$$y_z^m(t) + R_y(t - t_k) \le y_z(t) \le y_z^M(t) - R_y(t - t_k), \quad t \in [t_k, t_{k+C}], \tag{26}$$

with the requirement

$$0 \le R_y(t) \le d_{z,y}/2, \tag{27}$$

whereas the robust stability constraint is

$$\sum_{i=0}^{C-1} (e^{-i\eta\delta} - e^{-(i+1)\eta\delta}) \frac{\dot{y}_z^{k+i}}{\eta} = y_u^k - y_z^k (1 - e^{-\eta T_c}) - \tilde{c}_y^k + \tilde{c}_{w,y}^k, \tag{28}$$

where y_u is defined as

$$y_u = y_c + \dot{y}_c/\eta. \tag{29}$$

Assume now that at time t_k RIS-MPC is operating in standard mode, and let us define the following vectors collecting quantities over the control horizon:

$$\dot{X}_z^k = (\dot{x}_z^k, \dots, \dot{x}_z^{k+C-1}), \quad \dot{Y}_z^k = (\dot{y}_z^k, \dots, \dot{y}_z^{k+C-1}),$$

$$X_z^k = (x_z^{k+1}, \dots, x_z^{k+C}), \quad Y_z^k = (y_z^{k+1}, \dots, y_z^{k+C}),$$

$$X_{\mathrm{mc}}^k = (x_{\mathrm{mc}}^{k+1}, \dots, x_{\mathrm{mc}}^{k+C}), \quad Y_{\mathrm{mc}}^k = (y_{\mathrm{mc}}^{k+1}, \dots, y_{\mathrm{mc}}^{k+C}).$$

The decision variables $\dot{X}_z^k$ and $\dot{Y}_z^k$ are determined by solving the following QP problem, called QP-SM:

$$\begin{cases} \min\limits_{\dot{X}_z^k, \dot{Y}_z^k} \|\dot{X}_z^k\|^2 + \|\dot{Y}_z^k\|^2 + \mu_1 \left(\|X_z^k - X_{\mathrm{mc}}^k\|^2 + \|Y_z^k - Y_{\mathrm{mc}}^k\|^2 \right) \\ \quad \text{subject to :} \\ \quad \bullet \text{ restricted ZMP constraints (11) for } x \text{ and (26) for } y; \\ \quad \bullet \text{ robust stability constraints (24) for } x \text{ and (28) for } y, \end{cases}$$

where μ_1 weighs the second term, which is aimed at keeping the ZMP close to the center of the moving constraint.

Note that, although the ZMP constraint is formulated in continuous time, it is actually enforced only at $t_k, \dots, t_{k+C}$. Indeed, in view of the linearity of x_z and y_z within each sampling interval, satisfaction of the original continuous-time constraint is guaranteed.

Once QP-SM has been solved, the first sample $\dot{x}_z^k$ of the resulting control sequence is used to integrate the prediction model (5), where the disturbance w_x is replaced by its estimate $\tilde{w}_x$, and to compute a reference CoM x-position for time t_{k+1}; and similarly with $\dot{y}_z^k$. A foot reference pose for the same time instant is generated by sampling the (predefined) foot trajectory for the current step. These CoM and foot references are then sent to a kinematic controller, which computes joint velocity commands for the humanoid.

4. Feasibility properties of the standard mode

In this section, we analyze the feasibility properties of the standard mode. In particular, we first characterize the feasibility region for QP-SM, and then use this result to prove recursive feasibility of RIS-MPC in standard mode.

4.1. Feasibility region of QP-SM

The following proposition characterizes the feasibility of the standard mode.

Proposition 2. *Problem QP-SM is feasible at t_k if and only if $(x_u^k, y_u^k) \in \mathcal{F}_{\mathrm{S}}^k$, where*

$$\mathcal{F}_{\mathrm{S}}^k = \{(x_u^k, y_u^k) : x_{u,\mathrm{S}}^{k,m} \le x_u^k \le x_{u,\mathrm{S}}^{k,M}, y_{u,\mathrm{S}}^{k,m} \le y_u^k \le y_{u,\mathrm{S}}^{k,M}\}$$

and the bounds of the feasibility region are given by

$$x_{u,\mathrm{S}}^{k,m} = \eta \int_{t_k}^{t_{k+C}} e^{-\eta(\tau - t_k)} \left(x_z^m(\tau) + R_x(\tau - t_k)\right) d\tau + \tilde{c}_x^k - \frac{\tilde{w}_x^k}{\eta^2}$$

$$x_{u,\mathrm{S}}^{k,M} = \eta \int_{t_k}^{t_{k+C}} e^{-\eta(\tau - t_k)} \left(x_z^M(\tau) - R_x(\tau - t_k)\right) d\tau + \tilde{c}_x^k - \frac{\tilde{w}_x^k}{\eta^2}.$$

and similar expression for $y_{u,\mathrm{S}}^{k,m}$ and $y_{u,\mathrm{S}}^{k,M}$.

Proof. See Appendix. ■

Since x_u, y_u are related to the state variables of the prediction model through (13) and (29), the above expression of $\mathcal{F}_{\mathrm{S}}^k$ actually identifies the feasibility region in state space. Note that the effect of the disturbance is only to displace the feasibility region in the opposite direction to the estimated disturbance $(\tilde{w}_x^k, \tilde{w}_y^k)$. Moreover, we have

$$x_{u,\mathrm{S}}^{k,M} - x_{u,\mathrm{S}}^{k,m} = d_{z,x}(1 - e^{-\eta T_c}) - 2\eta \int_0^{T_c} e^{-\eta\tau} R_x(\tau) d\tau, \tag{30}$$

showing how the extension of $\mathcal{F}_{\mathrm{S}}^k$ along x_u, which depends on the dimension $d_{z,x}$ of the ZMP admissible region, is reduced in the presence of a restriction function $R_x(\cdot)$; a similar result holds for the extension of $\mathcal{F}_{\mathrm{S}}^k$ along y_u.

An example relative to a forward walk is considered in Fig. 5, where the feasibility region $\mathcal{F}_{\mathrm{S}}$ of QP-SM is shown for a sequence of instants corresponding to the start of consecutive single support phases. The region is computed assuming $d_{z,x} = 0.1$ m, $\bar{z}_c = 0.7$ m, $T_c = 0.7$ s, $T_p = 2$ s, and a constant estimated disturbance $\tilde{w}_y = 0.6$ m/s^2. Both restrictions $R_x(\cdot)$ and $R_y(\cdot)$ are chosen to be linear functions (as in Fig. 4) having the maximum slope compatible with (12). For comparison, also shown are the feasibility regions $\mathcal{F}$ obtained by setting $R_x(\cdot) = R_y(\cdot) \equiv 0$ in (no restriction) and $\tilde{w}_x = \tilde{w}_y \equiv 0$ (no estimated disturbance in the stability constraints). As expected, each $\mathcal{F}_{\mathrm{S}}$ is displaced and reduced with respect to the corresponding $\mathcal{F}$.

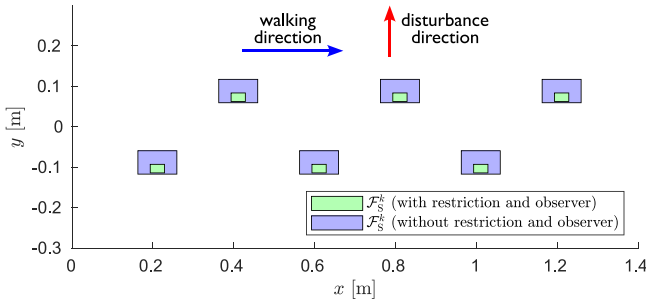


Fig. 5. The feasibility regions in the $x_u y_u$-plane at the starting instants of consecutive steps in a forward walk. The restriction of the ZMP constraints reduces the size of $\mathcal{F}_{\mathrm{S}}$ with respect to that of $\mathcal{F}$, whereas the inclusion of the observed disturbance value in the stability constraint results in a displacement between the two regions.

4.2. Recursive feasibility of RIS-MPC in standard mode

Based on the feasibility analysis in Proposition 2, we can prove the following result.

Proposition 3. *In the presence of perturbations belonging to $\mathcal{W}$, the proposed RIS-MPC scheme is recursively feasible in standard mode if*

$$\beta_x \ge \phi_x + \psi_x \tag{31}$$

$$\beta_y \ge \phi_y + \psi_y, \tag{32}$$

where

$$\beta_i = -\eta \int_0^{\delta} e^{-\eta\tau} R_i(\tau) d\tau + \eta(e^{\eta\delta} - 1) \int_{\delta}^{T_c} e^{-\eta\tau} R_i(\tau) d\tau \tag{33}$$

for $i \in \{x, y\}$, and correspondingly

$$\phi_i = e^{-\eta T_c}(e^{\eta\delta} - 1) \left(\frac{e^{-\eta(T_p - T_c)}}{\eta} v_{z,i}^{\max} - \frac{1}{2} d_{z,i} \right) \tag{34}$$

$$\psi_i = \frac{2\Delta_i(e^{\eta\delta} - 1) + w_i^{\mathrm{diff}}}{\eta^2}, \tag{35}$$

with $v_{z,x}^{\max}$, $v_{z,y}^{\max}$ respectively the maximum x- and y-velocity of the ZMP in the anticipative tail.

Proof. See Appendix. ■

We now provide a few comments on the interpretation of Proposition 3, proceeding from the basic observation that in view of the sufficient conditions (31)–(32) one would like to make β_i as large as possible and ϕ_i, ψ_i as small as possible, for $i \in \{x, y\}$.

- Each term β_i depends on the corresponding restriction function $R_i(\cdot)$. In particular, according to (33), β_i decreases with the value of $R_i(t)$ in $[0, \delta]$ and increases with its value in $[\delta, T_c]$.
- Formula (34) shows that ϕ_i increases with $v_{z,i}^{\max}$ (which is exogenous, as it depends on the footstep plan) and decreases with T_c (longer prediction horizons help), with $T_p - T_c$ (benefit of the preview information, consistently with the findings in [27, Prop. 5]), and with the corresponding dimension of the footprint.
- Formula (35) shows that ψ_i increases with the maximum deviation Δ_i of the disturbance (while its midrange value does not matter), as well as with the bound w_i^{diff} on the increment between consecutive estimates of w_i.
- Assume that the disturbance is constant and known. In this case, we have $\Delta_i = 0$, so that $w_i^{\mathrm{diff}} = 0$, and the robust stability constraint incorporates the exact value of w_i from t_k to ∞, effectively performing an indirect disturbance compensation (we will come back to this in Section 11.3). Now, remove the restriction function ($R_i(\cdot) \equiv 0$) in the ZMP constraints. Then, it is easy to verify that the recursive feasibility conditions (31)–(32) for robust gait generation reduce to those of the nominal case, see [27, Eq. (28)].

An upper bound on the parameter β_i, $i \in \{x, y\}$, in Proposition 3 is found by letting $R_i(t) = 0$ in $[0, \delta]$ and, in view of requirements (12) and (27), $R_i(t) = d_{z,i}/2$ in $[\delta, T_c]$. One obtains

$$\beta_i^{\max} = \frac{d_{z,i}}{2}(e^{-\eta\delta} - e^{-\eta T_c})(e^{\eta\delta} - 1).$$

As a consequence, the proposed RIS-MPC scheme guarantees recursive feasibility in standard mode for disturbances such that

$$\psi_i \leq \beta_i^{\max} - \phi_i = (e^{\eta\delta} - 1)\left(\frac{d_{z,i}}{2}e^{-\eta\delta} - \frac{e^{-\eta T_p}}{\eta}v_{z,i}^{\max}\right),$$

that is

$$\Delta_i \leq \frac{\eta^2}{2}\left(\frac{d_{z,i}}{2}e^{-\eta\delta} - \frac{e^{-\eta T_p}}{\eta}v_{z,i}^{\max}\right) - \frac{w_i^{\text{diff}}}{2(e^{\eta\delta} - 1)}. \tag{36}$$

However, it should be emphasized that (31)–(32) are only sufficient conditions for recursive feasibility. Indeed, less restrictive conditions are obtained by replacing $2\Delta_i$ in (35) with $\max_{t\in[t_k, t_{k+1}]} |w_x(t) - \tilde{w}_x^k|$, i.e., the maximum estimation error (see the proof of Proposition 3 in the Appendix, and in particular Eq. (53)). Depending on the performance of the estimator, these less restrictive conditions may be satisfied even when (31)–(32) are violated, as will be shown via simulation in Section 5.

4.3. Feasibility-driven design of the restriction

Proposition 3 indicates that a restriction of the ZMP constraints is beneficial for recursive feasibility in standard mode. However, it also reduces the size of the feasibility region, as shown by Eq. (30) and the equivalent expression for the extension of $\mathcal{F}_S^k$ along y_u. Our objective is therefore to design $R_x(\cdot)$, $R_y(\cdot)$ in such a way that an acceptable trade-off is achieved between these two conflicting aspects. In the following, we refer to the design of $R_x(\cdot)$ for illustration; similar arguments and formulas apply for $R_y(\cdot)$.

Define the *feasibility decrease* D_x in x_u as the difference between the extension of $\mathcal{F}_S^k$ along x_u in the unrestricted and the restricted case. Using Eq. (30) first with $R(t) = 0$ and then with $R(t) \neq 0$, one easily finds

$$D_x = 2\eta \int_0^{T_c} e^{-\eta\tau} R_x(\tau) d\tau. \tag{37}$$

We would like to define $R_x(\cdot)$ so as to minimize D_x while satisfying the sufficient condition (31) for recursive feasibility. In particular, we will search within a specific family of restriction functions defined as follows:

$$R_x(t) = \begin{cases} 0 & \text{for } t \in [0, \delta) \\ \frac{d_{z,x}}{2}\left(1 - e^{-\rho_x(t-\delta)}\right) & \text{for } t \in [\delta, T_c], \end{cases} \tag{38}$$

This profile of $R_x(\cdot)$ satisfies requirement (12) and is consistent with the previous discussion on the influence of the restriction function on β_x: in particular, the function goes exponentially, with rate ρ_x, from its zero value in $[0, \delta)$ to the maximum admissible value $d_{x,z}/2$. The corresponding value of D_x is obtained from (37) as

$$D_x = d_{z,x}\left(e^{-\eta\delta} - e^{-\eta T_c} + \frac{\eta\, e^{\rho_x\delta}}{\eta + \rho_x}\left(e^{-(\eta+\rho_x)T_c} - e^{(\eta+\rho_x)\delta}\right)\right). \tag{39}$$

At this point, designing the restriction function boils down to solving the following constrained optimization problem:

$$\min_{\rho_x} D_x \qquad \text{subject to} \quad \beta_x \geq \phi_x + \psi_x.$$

Using (38) in (33), with $i = x$, we obtain

$$\beta_x = \frac{e^{\eta\delta} - 1}{2} D_x.$$

Therefore, to minimize D_x one must choose the smallest β_x such that $\beta_x \geq \phi_x + \psi_x$; i.e., the latter constraint must be active at the optimum.

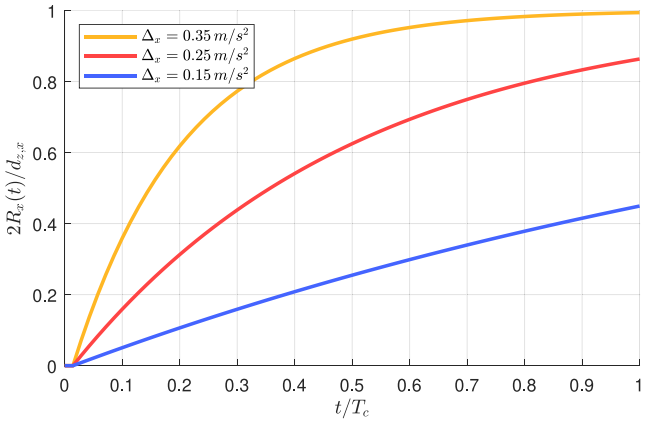


Fig. 6. The optimal restriction function in the family (38) for different values of the maximum deviation Δ_x of the disturbance. The plot is normalized with respect to $d_{z,x}/2$.

This means that the optimal ρ_x can be directly found by solving numerically the constraint, which using (39) is expressed as:

$$\frac{\eta}{\eta + \rho_x} e^{\rho_x\delta}\left(e^{-(\eta+\rho_x)\delta} - e^{-(\eta+\rho_x)T_c}\right) = \nu, \tag{40}$$

with the constant term ν given by

$$\nu = e^{-\eta\delta} - e^{-\eta T_c} - \frac{2(\phi_x + \psi_x)}{d_{z,x}(e^{\eta\delta} - 1)}.$$

One may easily verify that the left-hand side of (40) is a monotonically decreasing function of ρ_x, and therefore whenever a solution exists[5] it is unique.

In practice, the optimal restriction function along x is computed in advance via (40) given the parameters of the robot (η, $d_{z,x}$), of the MPC (δ, T_c), of the footstep plan ($v_{z,x}^{\max}$, T_p), of the disturbance (Δ_x) and of the disturbance estimator (w_x^{diff}).

Some restriction functions generated with the above procedure are shown in Fig. 6. Here, we set $\eta = 3.74$ Hz, $d_{z,x} = 0.1$ m, $\delta_x = 0.01$ s, $T_c = 0.7$ s, $T_p = 1$ s, $v_{z,x}^{\max} = 0.68$ m/s, $w_x^{\text{diff}} = 0.45$ m/s^2, and we considered three different values for the maximum deviation Δ_x of the disturbance, corresponding to 0.15, 0.25 and 0.35 m/s^2. As expected from (38), all three functions are zero in the first sampling interval and exponentially converge to $d_{z,x}/2$. The optimal rate of convergence ρ_x increases with Δ_x, consistently with Eq. (40), whose right-hand side is a decreasing function of Δ_x through ψ_x. In particular, for the considered values of Δ_x we find $\rho_x = 0.86$, 2.88 and 7.40.

The above results confirm that the restriction functions $R_x(\cdot)$, $R_y(\cdot)$ must become more drastic as the maximum deviations Δ_x, Δ_y of the disturbance increase. Correspondingly, the area of the feasibility region $\mathcal{F}_S^k$ will decrease.

5. Standard mode: LIP simulations

In this section, MATLAB simulations will be presented for the perturbed LIP model in order to illustrate the performance of the RIS-MPC scheme in standard mode.

The following parameters are used:

[5] Note that the proposed family (38) includes, as a limit for $\rho_x \to \infty$, the strongest possible restriction function, i.e., $R_x(t) = d_{z,x}/2$ for $t \in [\delta, T_c]$. Therefore, looking for the optimal restriction function within family (38) is not restrictive with respect to feasibility. If no solution ρ_x exist for Eq. (40), one can conclude that no restriction function can satisfy the sufficient conditions for recursive feasibility under the considered disturbance set $\mathcal{W}$.

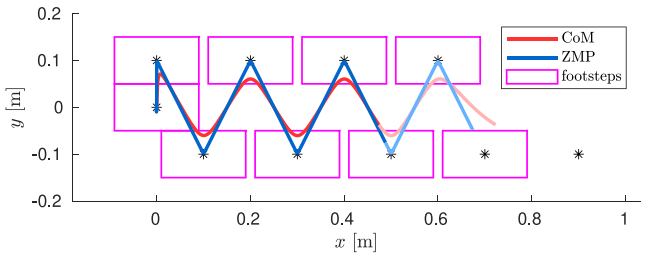


Fig. 7. LIP Simulation 1: The gait generated by IS-MPC in the absence of disturbances.

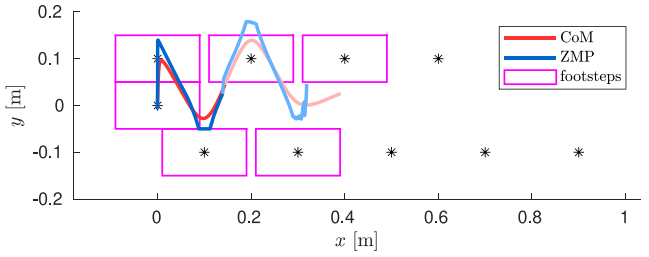


Fig. 8. LIP Simulation 2: The gait generated by IS-MPC in the presence of a constant disturbance $w_y = 0.2$ m/s^2.

- For the robot, we set $d_{z,x} = 0.18$ m, $d_{z,y} = 0.10$ m (footprint dimensions), $\bar{z}_c = 0.75$ m (CoM height), $d_{a,x} = 0.25$ m, $d_{a,y} = 0.07$ m, $\ell = 0.23$ m (kinematically admissible area).
- For RIS-MPC, we set $\delta = 0.01$ s (sampling interval), $T_c = 2$ s (control horizon) and $T_p = 4$ s (preview horizon). In QP-SM, we chose $\mu_1 = 100$. The reduced-order observer used for estimating the disturbance is chosen in such a way that the eigenvalues of the reduced observation error dynamics are in $\{-10, -5(1 \pm j\sqrt{3})\}$ (Butterworth configuration), while the w_{diff} parameter of the saturator is set to 0.003 m/s^2 for both x and y (the other parameter Δ will instead depend on the considered disturbance set). All QPs are solved with the MATLAB `quadprog` function.

These parameters will be unchanged throughout the paper, except when explicitly specified. Note that the robot data are appropriate for the HRP-4 humanoid platform.

The rest of the parameters, which are related to the footstep plan and the disturbance, are specific to this first group of simulations.

The footsteps are equally spaced by 0.1 m in the x direction and alternatively displaced by ± 0.1 m in the y direction, with a step duration of $T_s = 1$ s (correspondingly, $F = 2$), divided in a double support phase of $T_{ds} = 0.4$ s and a single support phase of $T_{ss} = 0.6$ s. The maximum ZMP velocities are therefore $v_{z,x}^{\max} = 0.1$ m/s and $v_{z,y}^{\max} = 0.2$ m/s.

The expected disturbance sets $\mathcal{W}_x$ and $\mathcal{W}_y$ are defined by $w_{m,x} = w_{m,y} = 0$ m/s^2 (midrange) and $\Delta_x = \Delta_y = 0.2$ m/s^2 (maximum deviation). For the HRP-4 robot, whose mass is 40 kg, the forces equivalent to the maximum disturbance in the two directions would be 8 N.

In Simulation 1, there is no disturbance acting on the LIP ($w_x = w_y = 0$), and gait is generated using the IS-MPC method of [27]. Fig. 7 shows[6] the resulting unperturbed gait, with the CoM trajectory centered with respect to the footsteps.

In Simulation 2, we set $w_x = 0$ m/s^2 and $w_y = 0.2$ m/s^2, i.e., a constant push on the CoM coming from the right. IS-MPC now fails, producing a gait where the CoM starts drifting toward the left footsteps because feasibility is lost (Fig. 8).

[6] In all COM-ZMP plots, the lighter lines indicate the prediction within the control horizon.

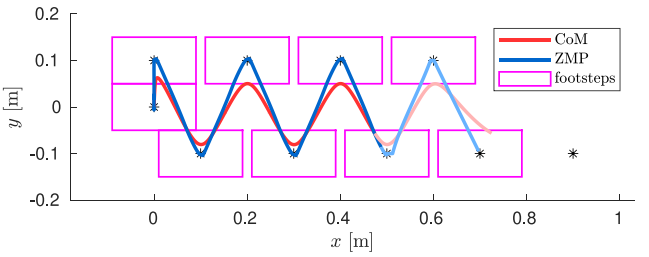


Fig. 9. LIP Simulation 3: The gait generated by RIS-MPC in the presence of a constant disturbance $w_y = 0.2$ m/s^2.

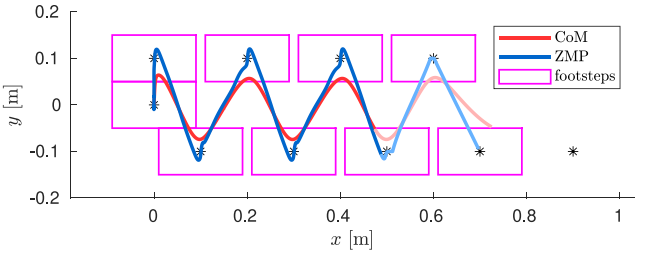


Fig. 10. LIP Simulation 4: The gait generated by RIS-MPC in the presence of a triangular-wave disturbance w_y.

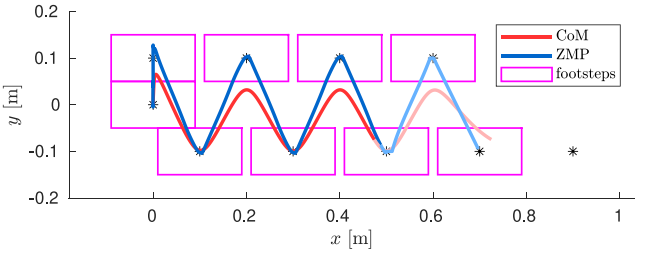


Fig. 11. LIP Simulation 5: The gait generated by RIS-MPC in the presence of a constant disturbance $w_y = 0.44$ m/s^2.

In Simulation 3, we still assume $w_y = 0.2$ m/s^2 but gait generation is now obtained by the RIS-MPC method. An optimal restriction function R_y which guarantees recursive feasibility is computed from the expected disturbance set as discussed in Section 4.3, leading to $\rho_x = 2.67$ and $\rho_y = 11.71$. As expected, feasibility is maintained throughout the simulation, so that the scheme continuously operates in standard mode. The resulting gait (Fig. 9) is stable and exhibits a clear displacement of the CoM towards the right, i.e., the direction from which the disturbance is coming. Interestingly, this behavior can be interpreted as the humanoid *leaning against the disturbance* in order to counteract it; we will come back to this in Section 11.3.

Simulation 4 is aimed at showing the performance of the method in the presence of a time-varying disturbance in $\mathcal{W}_y$. In particular, it is assumed that w_y is a triangular wave with zero initial value, period of 1 s and peak value 0.2 m/s^2; this means that the disturbance still comes from the right but its intensity oscillates around a nonzero value. As a consequence, the resulting gait, shown in Fig. 10, is characterized by a slightly smaller displacement of the CoM towards the disturbance with respect to the previous simulation.

In Simulation 5, we go back to a constant lateral disturbance, but we increase its intensity to $w_y = 0.44$ m/s^2, which is outside the assigned set $\mathcal{W}_y$, in order to prove that RIS-MPC can work even when the sufficient condition for recursive feasibility of Proposition 3 is violated. In fact, as mentioned at the end of Section 4.2, recursive feasibility can still be achieved at runtime if the disturbance estimation error remains sufficiently small; to this end, the saturator was removed, to allow the

estimator to accurately track the disturbance, also outside $\mathcal{W}_y$. Indeed, the results shown in Fig. 11 confirm that the proposed method is able to produce a stable gait also in this situation, with the humanoid leaning against the disturbance even more than in Simulation 3.

Animations of the above simulations are shown in the accompanying video.

6. Recovery mode

According to Fig. 1, the proposed method will automatically switch to recovery mode whenever QP-SM becomes unfeasible; to this end, the algorithm checks whether $(x_u^k, y_u^k) \in \mathcal{F}_S^k$ (see Proposition 2). Note that, if the sufficient conditions (31)–(32) for recursive feasibility have been guaranteed by design, QP-SM can become unfeasible only when the disturbance (w_x, w_y) goes outside[7] the known set $\mathcal{W}$, e.g., due to an impulsive component.

As mentioned in Section 2, the QP to be solved in recovery mode (QP-RM) is modified with respect to that of the standard mode in order to enlarge the feasibility region as much as possible. Moreover, a *preliminary* adaptation of step timings is performed (if needed) to make sure that such region includes the current state.

In this section, we describe in detail QP-RM and the step timing adaptation technique.

6.1. Recovery mode QP

In recovery mode, RIS-MPC can modify both the positions and timings of the footsteps with respect to those contained in the original plan. For the sake of clarity, we will denote the latter as

$$\hat{X}_f^k = (\hat{x}_f^1, \dots, \hat{x}_f^F), \quad \hat{Y}_f^k = (\hat{y}_f^1, \dots, \hat{y}_f^F), \quad \hat{\tau}_s^k = (\hat{t}_s^1, \dots, \hat{t}_s^F).$$

The decision variables of QP-RM are the ZMP velocity inputs and the footstep positions inside the control horizon:

$$\dot{X}_z^k = (\dot{x}_z^k, \dots, \dot{x}_z^{k+C-1}), \quad \dot{Y}_z^k = (\dot{y}_z^k, \dots, \dot{y}_z^{k+C-1}),$$
$$X_f^k = (x_f^1, \dots, x_f^F), \quad Y_f^k = (y_f^1, \dots, y_f^F),$$

whereas the adapted timings τ_s^k have already been selected by the STA module.

The ZMP constraints are expressed analogously to (10) as

$$x_z^m(t, X_f^k) \le x_z(t) \le x_z^M(t, X_f^k) \tag{41}$$
$$y_z^m(t, Y_f^k) \le y_z(t) \le y_z^M(t, Y_f^k) \tag{42}$$

having highlighted the dependence of the bounds on the adapted footsteps as these are now decision variables. The ZMP constraints in recovery mode are not restricted in order to maximize the probability that the resulting QP-RM is feasible.

To guarantee that the adapted footsteps are admissible for the robot, two *kinematic constraints* are introduced. Along x, this simply translates to enforcing a maximum step length:

$$|x_f^j - x_f^{j-1}| \le \frac{d_{a,x}}{2}, \tag{43}$$

with $d_{a,x}$ the size of the admissible region along x. Along y, a minimum and maximum lateral displacement are enforced:

$$\left| y_f^j - y_f^{j-1} \pm \ell \right| \le \frac{d_{a,y}}{2}, \tag{44}$$

where ℓ is the central value of the distance between right and left feet, and the sign alternates for the two feet.

Finally, the robust stability constraint (24) is formally unchanged in QP-RM. However, the footstep plan after the control horizon, which is used to form the anticipative tail needed to compute $\tilde{c}_x^k$, is rigidly displaced from the original plan to make it kinematically reachable from the last footstep in the control horizon (which is now a decision variable, and therefore different from the planned footstep). See Section 6.4 for more details.

Wrapping up, in recovery mode the following QP problem, called QP-RM, is solved:

$$\begin{cases} \min\limits_{\substack{\dot{X}_z^k, \dot{Y}_z^k \\ X_f^k, Y_f^k}} \|\dot{X}_z^k\|^2 + \|\dot{Y}_z^k\|^2 + \mu_1 \left(\|X_z^k - X_{\text{mc}}^k\|^2 + \|Y_z^k - Y_{\text{mc}}^k\|^2 \right) \\ \qquad + \mu_2 \left(\|X_f^k - \hat{X}_f^k\|^2 + \|Y_f^k - \hat{Y}_f^k\|^2 \right) \\ \text{subject to:} \\ \quad \bullet \text{ ZMP constraints (41) for } x \text{ and (42) for } y; \\ \quad \bullet \text{ kinematic constraints (43)–(44), for } j = 1, \dots, F; \\ \quad \bullet \text{ robust stability constraints (24) for } x \text{ and (28) for } y. \end{cases}$$

With respect to QP-SM, the cost function now includes an additional term (weighted by μ_2) aimed at minimizing the variation of the footstep plan with respect to the original.

6.2. Feasibility region of QP-RM

The step timing adaptation technique relies on the characterization of the feasibility region of QP-RM. To this end, we extend the result of [30] to the present case, in which the stability constraint includes an estimate of the disturbance.

Let

$$\Delta\lambda = e^{-\eta(t_s^1 - t_k)},$$

be an exponential encoding of the remaining duration of the current step, referred to as the *time-to-step*. This is the only variable that will be modified by the STA module, while the duration of the subsequent steps is left unchanged. The motivation behind this choice is twofold: on the one side, computations will be simpler; on the other, it is expected that the timing of the current step is the most crucial for absorbing a disturbance.

Proposition 4. *Problem QP-RM is feasible at t_k if $(x_u^k, y_u^k) \in \mathcal{F}_{\text{R,est}}^k$, with*

$$\mathcal{F}_{\text{R,est}}^k = \{(x_u, y_u) : \bar{x}_u^{k,m} \le x_u \le \bar{x}_u^{k,M}, \bar{y}_u^{k,m} \le y_u \le \bar{y}_u^{k,M}\},$$

where

$$\bar{x}_u^{k,m} = \alpha_x^m \Delta\lambda + \gamma_x^m - \frac{\tilde{w}_x^k}{\eta^2}$$
$$\bar{x}_u^{k,M} = \alpha_x^M \Delta\lambda + \gamma_x^M - \frac{\tilde{w}_x^k}{\eta^2},$$

and similar expressions for $\bar{y}_u^{k,m}$ and $\bar{y}_u^{k,M}$. Coefficients α_x^m, α_x^M, γ_x^m and γ_x^M take different expressions in single and double support; these can be found in [30].

Proof. See Appendix. ■

6.3. Step timing adaptation (STA)

Proposition 4 gives a sufficient condition for the feasibility of QP-RM, expressed through a conservative estimate $\mathcal{F}_{\text{R,est}}^k$ of the actual feasibility region. Exploiting the linearity in $\Delta\lambda$ of the bounds of $\mathcal{F}_{\text{R,est}}^k$, we can formulate the following QP for step timing adaptation (QP-STA),

[7] The reverse is not true: QP-SM may remain feasible even if the disturbance leaves the set $\mathcal{W}$.

to be solved before invoking QP-RM:

$$\begin{cases} \min_{\Delta\lambda} (\Delta\lambda - \widehat{\Delta\lambda})^2 \\ \text{subject to:} \\ \bullet \text{ feasibility constraints for } x_u \text{ and } y_u \\ \quad \bar{x}_u^{k,m} \le x_u \le \bar{x}_u^{k,M} \\ \quad \bar{y}_u^{k,m} \le y_u \le \bar{y}_u^{k,M}, \\ \bullet \text{ timing constraints} \\ \quad e^{-\eta(t_s^0 + T_s^{\max} - t_k)} \le \Delta\lambda \le e^{-\eta(t_s^0 + T_s^{\min} - t_k)} \\ \quad \Delta\lambda \le e^{-\eta\epsilon_t}, \end{cases}$$

where $\widehat{\Delta\lambda} = e^{-\eta(\hat{t}_s^1 - t_k)}$ is the exponential encoding of the time-to-step according to the candidate timing and

$$\epsilon_t = \begin{cases} \hat{T}_{ss}^0 + \max\left(|x_{\text{mc}}^k - x_f^0|/v_{z,x}^{\max}, |y_{\text{mc}}^k - y_f^0|/v_{z,y}^{\max}\right) & \text{(DS)} \\ \max\left(|x_{\text{sw}}^k - x_f^{1|k-1}|/v_{\text{sw},x}^{\max}, |y_{\text{sw}}^k - y_f^{1|k-1}|/v_{\text{sw},y}^{\max}\right) & \text{(SS)} \end{cases}$$

where $\hat{T}_{ss}^0$ is the candidate duration of single support in the current step, $(x_f^{1|k-1}, y_f^{1|k-1})$ is the next footstep as generated by the previous RIS-MPC iteration, and $(x_{\text{sw}}^k, y_{\text{sw}}^k)$ is the swing foot position at t_k.

The first timing constraint guarantees that the new step duration will be within $T_s^{\min}$ and $T_s^{\max}$, limit values that typically depend on the robot. The second entails that only the duration of the current phase (SS or DS) is adapted, so that in DS the time-to-step cannot be reduced below the candidate duration of SS; it also guarantees that the robot has enough time to complete the current phase, given the bounds on the ZMP velocity (in DS) and on the swinging foot velocity (in SS).

After solving QP-STA, the adapted timestamp of the next step is computed as $t_s^1 = t_k + \log(1/\Delta\lambda)/\eta$. Clearly, no adaptation will be necessary ($\Delta\lambda = \widehat{\Delta\lambda}$) if (x_u^k, y_u^k) belongs to $\mathcal{F}_{\text{R,est}}^k$.

In practice, the effect of adaptation will be dependent on the direction of the perturbation. For example, a push in the x direction (positive or negative) will be handled by reducing the current step duration so as to enlarge $\mathcal{F}_{\text{R,est}}^k$ along x. The same kind of adaptation will work for a lateral push directed from the support towards the swing leg (positive y), because $\mathcal{F}_{\text{R,est}}^k$ will be displaced in the direction of the push. If the lateral push is directed from the swing towards the support leg (negative y), the step duration will instead be increased, because this will shift $\mathcal{F}_{\text{est}}^k$ down in y.

6.4. Footstep plan update

In recovery mode, adapting the next F footsteps (those falling in the control horizon) may displace them significantly from the original plan. When going back to standard mode, footsteps are once again regarded as fixed, and it might be kinematically impossible for the robot to reconcile the adapted footsteps with the original plan. To avoid this, after solving QP-RM the footstep plan is updated as follows (see Fig. 12):

(1) the F candidate footsteps $\hat{X}_f^k$, $\hat{Y}_f^k$, which fall in the control horizon, are replaced by the adapted footsteps resulting from the solution of QP-RM;
(2) all the successive footsteps (i.e., from the $(F+1)$-th footstep on) are rigidly displaced in such a way that the $(F+1)$-th footstep is kinematically reachable from the Fth footstep.

Clearly, this simple update procedure assumes that there are no obstacles in the workspace. In a more general case, an actual replanning from the current footstep would be needed to guarantee at the same time obstacle avoidance and kinematic admissibility.

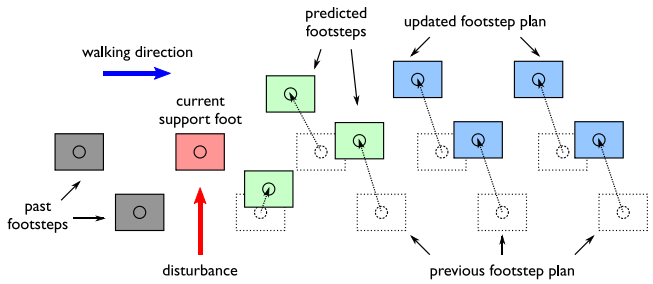


Fig. 12. Updating the footstep plan after solving QP-RM. Arrows indicate how each footstep is displaced from the original plan.

7. Complete RIS-MPC: LIP simulations

In this section, we show the performance of the complete RIS-MPC scheme in the presence of both persistent and impulsive disturbances acting on the LIP.

In Simulation 6, the robot parameters, the RIS-MPC parameters, the footstep plan and the expected disturbance sets $\mathcal{W}_x$, $\mathcal{W}_y$ are exactly the same of Section 5. In QP-RM, we set $\mu_1 = 100$ and $\mu_2 = 10000$. The lateral disturbance $w_y(t)$ consists of a constant term of 0.2 m/s^2 and an impulsive push of 18 m/s^2 at 4.1 s, while there is no disturbance in the sagittal direction ($w_x(t) = 0$).

The results are shown in Fig. 13. In the initial phase ($t = 3.60$ s), only the constant disturbance is present, and the scheme works in standard mode producing a gait leaning towards the right, identical to that of Simulation 3 (Fig. 9). When the effect of the impulsive disturbance appears ($t = 4.11$ s), the QP-SM problem becomes unfeasible, as shown by (x_u^k, y_u^k) being outside $\mathcal{F}_S^k$. The control algorithm then switches to recovery mode, whose feasibility region $\mathcal{F}_R^k$ is significantly larger and includes (x_u^k, y_u^k); this is obtained thanks to QP-STA, which reduces the step time from the nominal 1 s to 0.92 s. Note that, following step time adaptation, QP-RM modifies the footsteps with respect to the original plan. At $t = 4.53$ s, (x_u^k, y_u^k) is back in $\mathcal{F}_S^k$; this means that the effect of the impulsive push has been absorbed and standard mode can be resumed. Finally, at $t = 6.00$ s, the robot is back to the initial gait, with the footstep plan displaced with respect to the original as explained in Section 6.4.

An animation of the above simulation is shown in the accompanying video.

8. Kinematic control

When using a template model (in our case, the LIP) for humanoid gait generation, it is necessary to convert the obtained CoM trajectory to appropriate commands for the actual robot. Depending on the control architecture, these commands may be velocities/accelerations or torques. In the following, we consider the first case (*kinematic control*), but RIS-MPC can also be used with torque-controlled humanoids, see the related comments in Section 12.

In our implementation, the robot commands are the joint accelerations, which are computed by solving a constrained optimization problem. In particular, the decision variables at time t_k are $\ddot{q}^k = (\ddot{q}_f^k, \ddot{q}_j^k)$, where $\ddot{q}_f^k$ is the 6-vector of linear and angular accelerations of the robot floating base and $\ddot{q}_j^k$ is the n-vector of the joint accelerations, being n the number of actuated dof's.

The primary task is maintaining the contact between the support foot and the ground. This is simply expressed by the following constraint

$$J_{\text{sup}}^k \ddot{q}^k + \dot{J}_{\text{sup}}^k \dot{q}^k = 0, \tag{45}$$

where J_{sup}^k is the $6 \times n$ geometric Jacobian of the support foot computed at the current configuration.

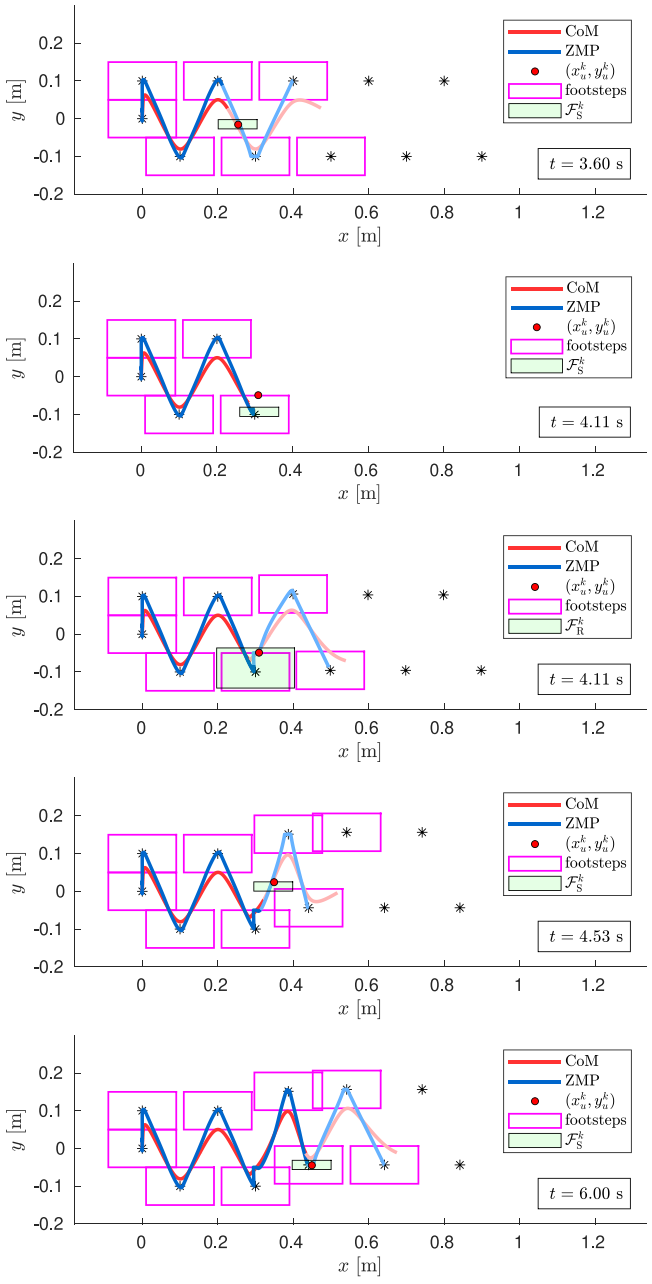


Fig. 13. LIP Simulation 6: The gait generated by RIS-MPC in the presence of a constant lateral disturbance of 0.2 m/s^2 with an added impulsive disturbance of 18 m/s^2 at 4.1 s.

The secondary task variables are defined by the 12-dimensional stack $p = (p_c, p_f)$, where p_c and p_f are the pose (6-dimensional position and orientation) of the frames attached to the CoM[8] and to the swinging foot, respectively. The reference for the secondary task is to track a reference p_{des}, defined as follows:

- $p_{c,\text{des}}$ consists of the desired position of the CoM frame, as generated by the currently active QP (either QP-SM or QP-RM), and its desired orientation, which is always specified as zero roll, pitch and yaw in our simulations and experiments.
- $p_{f,\text{des}}$ is computed by sampling a parameterized polynomial trajectory leading the pose of the current swing foot from its previous to the next landing configuration.

The acceleration vector $\ddot{q}^k$ at t_k is then computed by solving the following QP problem:

$$\begin{cases} \min_{\ddot{q}^k} \|J^k \ddot{q}^k + \dot{J}^k \dot{q}^k - \ddot{p}_{\text{des}}^k - K_p e_p^k - K_v \dot{e}_p^k\|_Q^2 \\ + \|S\,\ddot{q}^k - \ddot{q}_{\text{sel,des}}^k - K_{j,p} e_j^k - K_{j,v} \dot{e}_{j,p}^k\|^2 \\ \text{subject to:} \\ \bullet \text{ contact constraint (45)} \end{cases}$$

In this QP, the primary task is imposed as a hard constraint, while the remaining tasks are handled through the combined cost function. In the first term, which is aimed at realizing the secondary task for p, J is the task Jacobian, $e_p = p_{\text{des}} - p$ is the task error, Q is a weighting matrix, while K_p, K_v are positive definite gain matrices. The second term, which enforces tracking of a reference trajectory $q_{\text{sel,des}}$ for selected joints, is introduced to achieve additional tasks related to limb motion (mainly arm swinging) as well as to help solve kinematic redundancy (e.g., by specifying a target configuration); here, S is a selection matrix that isolates the relevant joints, e_j is the error with respect to the reference trajectory, and $K_{j,p}$ and $K_{j,v}$ are positive definite gain matrices.

After solving the above QP, the floating base accelerations $\ddot{q}_f^k$ are discarded, while the joint accelerations $\ddot{q}_j^k$ are commanded to the robot. In the case of a velocity- or position-controlled platform, the actual command can be reconstructed by integrating the accelerations once or twice, respectively.

9. Dynamic simulations on HRP-4

While the previous simulations were run on the LIP model, in this section we illustrate the performance of the proposed control scheme via dynamic simulations on the HRP-4 robot in the DART environment. See the accompanying video for complete animations of all the results.

The robot parameters, the RIS-MPC parameters and the expected disturbance sets are exactly the same of Sections 5 and 7. QPs are solved by means of the HPIPM C library [41].

In Dynamic Simulation 1, there is a persistent lateral force of 30 N applied to the robot waist. Fig. 14 shows the gait produced by the proposed control scheme operating in standard mode, with the robot clearly leaning against the disturbance (compare with LIP Simulation 3).

In Dynamic Simulation 2, an impulsive lateral push of 110 N is applied for 0.1 s to the robot waist. The RIS-MPC scheme handles the disturbance by switching to recovery mode, which results in adaptation of timing and footsteps (Fig. 15). In particular, the latter are permanently displaced following the footstep plan update.

In Dynamic Simulation 3, HRP-4 is carrying on its forearms a box-shaped object with a mass of 8 kg, unknown to the robot. As a consequence, the overall CoM is displaced and the weight of the system is increased. As shown in [42], this situation can be equivalently seen as a perturbation of the original LIP dynamics, namely an additional unknown horizontal force acting on the robot CoM from behind. The gait produced by RIS-MPC is shown in Fig. 16; note how the robot leans backward to counteract the disturbance. There is also a slight footstep adaptation (better appreciated in the video), indicating that the control scheme went temporarily in recovery mode; this is due to the object rolling on the robot arms, which introduces impulsive components in the disturbance.

The remaining simulations refer to other possible scenarios of interest (Fig. 17). In Dynamic Simulation 4, the humanoid is walking forward when its shoulder collides with a suspended shelf. RIS-MPC handles the corresponding impulsive disturbance by activating the

[8] The orientation of the CoM frame is assigned to the robot torso.

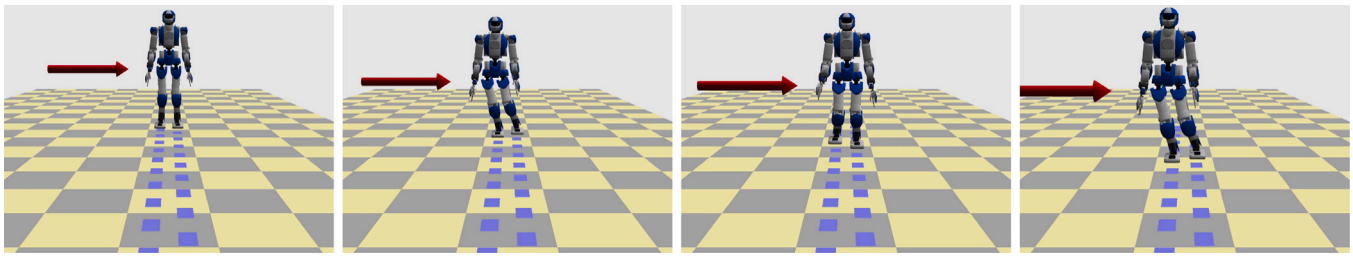


Fig. 14. HRP-4 Dynamic Simulation 1: Gait generated by RIS-MPC in the presence of a constant lateral force of 30 N acting on the robot waist. Note how the robot leans against the disturbance to counteract it.

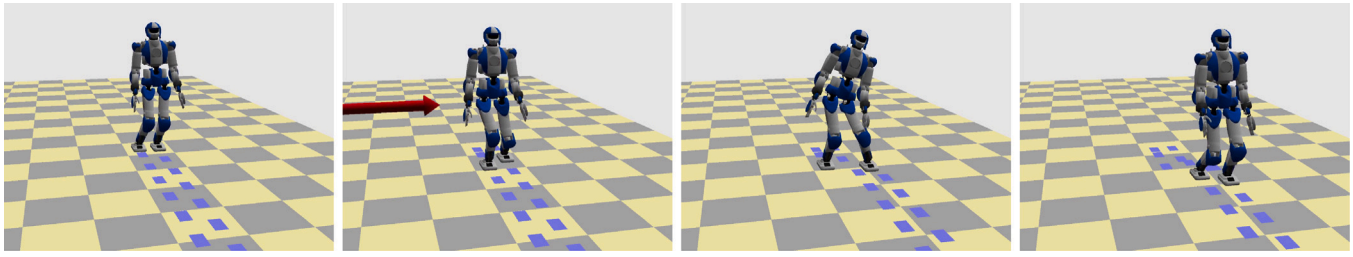


Fig. 15. HRP-4 Dynamic Simulation 2: Gait generated by RIS-MPC in the presence of an impulsive lateral force of 110 N acting for 0.1 s on the robot waist. After the disturbance is absorbed, the footstep plan is permanently displaced.

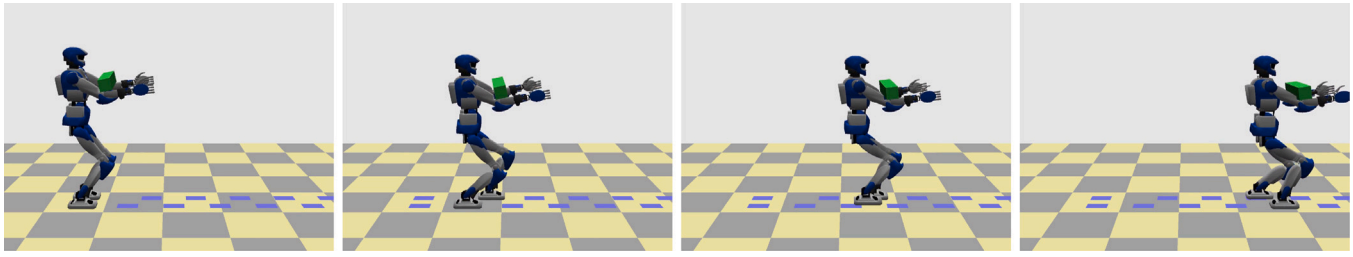


Fig. 16. HRP-4 Dynamic Simulation 3: Gait generated by RIS-MPC when the robot is carrying an object of 8 kg. In this case, the robot leans slightly backwards to counteract the force disturbance due to the weight of the object.

recovery mode and adapting both timings and footsteps. Once the effect of the perturbation has been absorbed, HRP-4 continues walking on the updated footstep plan (see Section 6.4).

In Dynamic Simulation 5, the robot is carrying a 1.5 kg object while walking down a ramp with different slopes, all unknown to the robot: 0°, 2.5°, 4°, 0° (Fig. 17, center). In this case, the proposed method is able to generate a robust gait using only the standard mode.

Finally, Dynamic Simulation 6 involves a cooperative transportation task where two HRP-4 robots walk while carrying a 16 kg bar (Fig. 17, bottom). The interaction forces arising during cooperative transportation can be regarded as disturbances on the individual CoMs [43]. Each robots runs an independent RIS-MPC algorithm, which effectively counteracts the perceived disturbance; ultimately, cooperative transportation of the bar is successfully accomplished. While this may not be the most sophisticated way of achieving cooperative transportation, the robustness achieved with the proposed method may prove beneficial in actual cooperation frameworks.

To give a more precise characterization of the disturbance rejection capability of RIS-MPC, we have also run a campaign of dynamic simulations with either persistent or impulsive (over 0.1 s) disturbances of increasing intensity along 12 uniformly distributed radial directions. All the parameters are the same of the previous dynamic simulations. The results are summarized in Fig. 18. The robustness gained with RIS-MPC in the case of persistent disturbances is remarkable, with an improvement of over 800% in certain directions. In the impulsive case, the improvement is less dramatic, ranging from 11 to 70% circa.

10. Experiments

To validate the proposed framework and show its versatility, we have conducted physical experiments on two small-size humanoids, i.e., a NAO by Aldebaran and an OP3 by Robotis. In both cases, we were able to implement the proposed control framework (RIS-MPC and kinematic control) fully on-board, using the qpOASES library to solve all the QPs. Due to the different computing power of the two platforms, RIS-MPC runs at 25 Hz on NAO and at 125 Hz on OP3; the QPs are solved using respectively qpOASES and HPIPM. In particular, the following are the computation times on OP3, which is the most challenging case: for QP-SM, mean value is 4.815 ms and standard deviation is 0.438 ms; for QP-RM, mean value is 5.770 ms and standard deviation is 1.237 ms; for QP-STA, mean value is 0.068 ms and standard deviation is 0.014 ms.

Four experiments were run on the NAO, see Fig. 19 and the accompanying video for full clips. Since the mass of NAO is 5.3 kg circa, the following robot parameters were used: $\bar{z}_c = 0.21$ m, $d_{z,x} = d_{z,y} = 0.05$ m, $d_{a,x} = 0.16$ m, $d_{a,y} = 0.05$ m, $\ell = 0.1$ m. For the MPC, we set $T_c = 1$ s and $T_p = 2$ s (consider that the nominal step duration was $\hat{T}_s = 0.5$ s in all experiments).

In the first two experiments, NAO walks down ramps whose slope is unknown to the controller; the associated gravity force is therefore treated as a disturbance. In the first experiment, there is also a variation of the slope (from 3° to 6°) along the ramp. RIS-MPC was always able to produce a stable gait, while IS-MPC could not (see the video for a comparison).

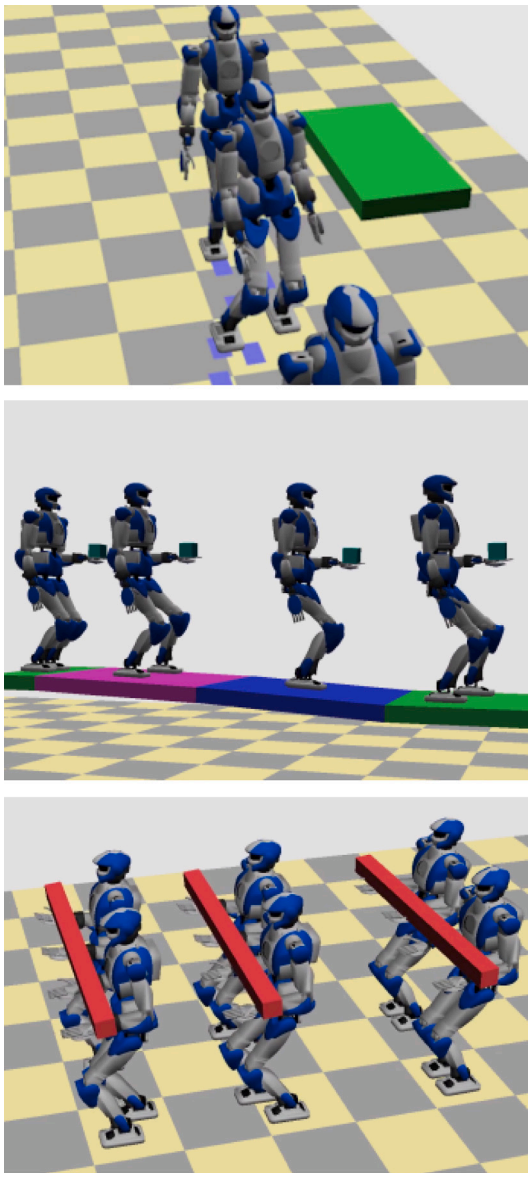


Fig. 17. HRP-4 Dynamic Simulations 4–6: Stroboscopic view of the gaits generated by RIS-MPC. Top: while the humanoid is walking, its shoulder collides with a fixed obstacle. Center: the humanoid walks down a ramp with two different slopes while carrying a 1.5 Kg cube. Bottom: two humanoids cooperate to carry a 16 Kg bar.

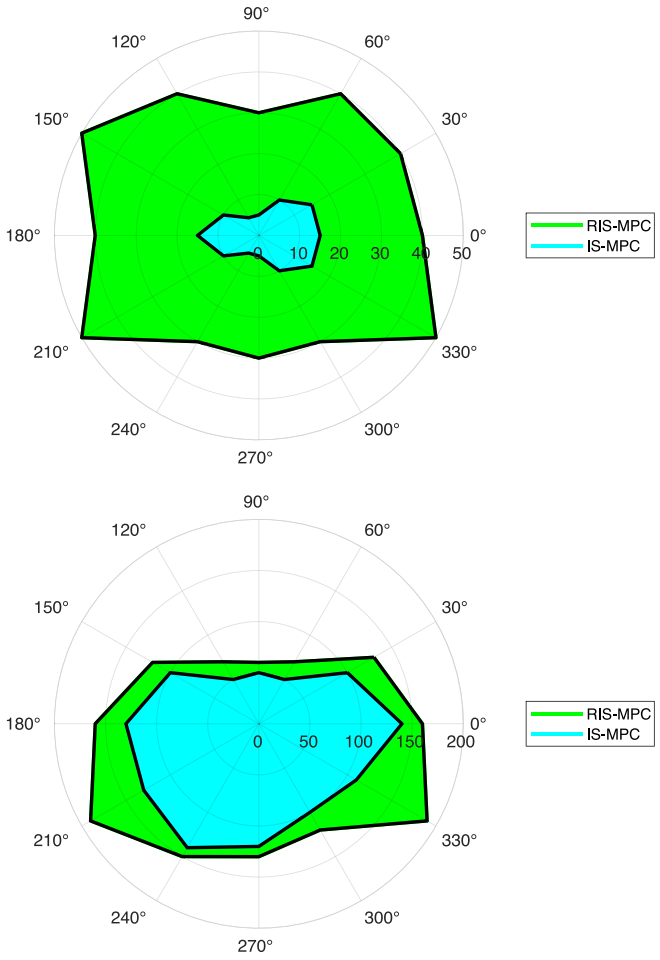


Fig. 18. The maximum disturbance (in N) that RIS-MPC and IS-MPC can withstand in the various directions: constant disturbances (top) and impulsive disturbances (bottom). HRP-4 is walking along the 0° direction.

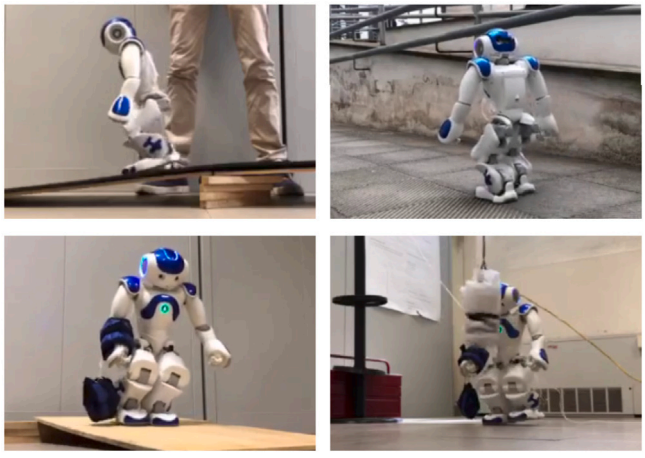


Fig. 19. NAO Experiments 1-4: walking down a ramp with two different slopes (top left), walking down an outdoor ramp (top right), walking down a ramp while carrying loads (bottom left), push from the front while carrying loads (bottom right) (see the accompanying video).

In Experiment 3, NAO is again descending a ramp of unknown slope, but two loads have been now added to the robot in order to introduce a further, asymmetric source of disturbance. In particular, the first load is rigidly attached to the robot right forearm, while the second is suspended from its right elbow (and repeatedly hits the right leg). The weight of both loads (circa 0.5 kg) is unknown to the controller. RIS-MPC was able to generate a stable gait in ten repeated trials. The measured CoM/ZMP trajectories obtained in one of these trials are shown in Fig. 20: note how the robot is leaning towards the direction from which the total disturbance is perceived. Instead, IS-MPC led to a fall of the robot in eight out of ten trials, with the other two resulting in a gait which strongly deviated from the forward direction.

In Experiment 4, NAO is walking forward carrying the same asymmetrical loads of the previous experiments, when it is hit from the front by a swinging pendulum of mass 0.67 kg. This experiment was repeated ten times with a 70% success rate for RIS-MPC. In particular, the impulsive disturbance triggered the activation of the recovery mode, which counteracts the disturbance by modifying both the timing and the footsteps. The measured COM/ZMP trajectories of a successful trial are shown in Fig. 21. In addition to the leaning effect, also evident here is the adaptation of the footsteps in response to the push.

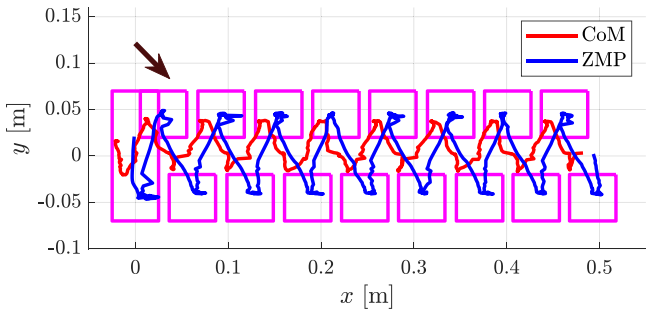


Fig. 20. NAO Experiment 3 (walking down a ramp while carrying loads): CoM and ZMP trajectories. The robot walks from the left to the right. The arrow indicates the direction of the total disturbance, due to the unknown slope and loads.

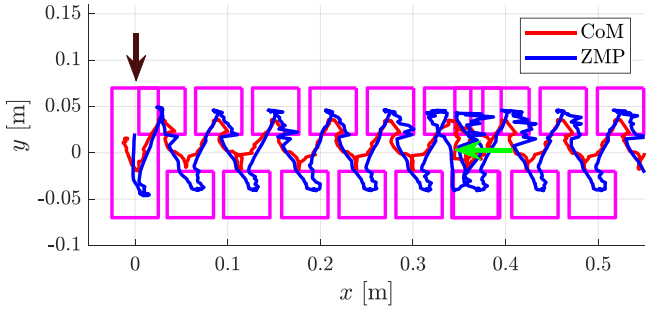


Fig. 21. NAO Experiment 4 (push from the front while carrying a load): CoM and ZMP trajectories. The robot walks from the left to the right. The black arrow indicates the direction of the disturbance due to the loads, while the green arrow represents the direction of the push. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

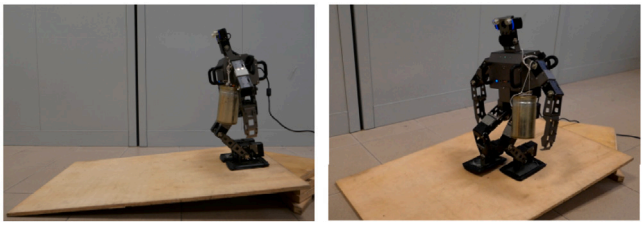


Fig. 22. OP3 Experiment: walking down a ramp while carrying a load (see the accompanying video).

For the experiments on the Robotis OP3 humanoid, the only different robot parameter with respect to the NAO implementation was $\bar{z}_c = 0.205$ m; all the other parameters were unchanged. For the MPC, we set $T_c = 1.3$ s, $T_p = 2.6$ s, for a nominal step duration $\hat{T}_s = 0.65$ s.

We present results from a single experiment on OP3, very similar to in nature to NAO Experiment 3. The OP3 robot, whose mass is around 3.5 kg, must walk down a ramp of unknown slope (−6°) while carrying a 0.3 kg weight suspended from its left shoulder, see Fig. 22. Also in this case, RIS-MPC was able to generate a stable gait in all trials, whereas the application of IS-MPC invariably led to failure.

11. Discussion

Having described the proposed scheme and showcased its performance, we discuss below a few aspects which deserve a more in-depth treatment.

11.1. Why two modes of operation?

One of the features of the proposed architecture is the separation between standard and recovery mode. One may wonder whether these could be merged into a single mode, with footstep positions always considered as decision variables. In such single-mode version of the scheme, footstep adaptation would be occurring at all times, and the cost function should then include a term aimed at keeping the adapted footsteps close to the candidates as much as possible (as in the cost function of QP-RM). In our experience, tuning the weight of this term proves very challenging, often resulting in the robot either adapting footsteps all the time (even when it would not be needed) or not adapting them at all, and consequently losing feasibility more often. Conversely, with the two modes separated, the footsteps are only modified in recovery mode; tuning the associated cost term becomes then much easier, because this mode invariably needs adaptation to reclaim feasibility.

Another benefit of the dual-mode architecture is that recursive feasibility of the standard mode can be proven under suitable assumptions on the disturbance (see Proposition 3), but generic disturbances (e.g., impulsive pushes) will still be handled thanks to the recovery mode. In a single-mode architecture, it would be impossible to achieve any theoretical guarantee of feasibility while simultaneously considering all kinds of disturbances; indeed, this is a common concern with robust control approaches, which can be overly conservative, resulting in a severely restricted range of operation.

11.2. Control flow within RIS-MPC

The RIS-MPC method operates in standard mode as long as the disturbance (w_x, w_y) belongs to the expected disturbance set $\mathcal{W}$. In fact, Proposition 3 implies that recursive feasibility of QP-SM is guaranteed, since the restriction function has been chosen so as to satisfy (31)–(32).

If the disturbance leaves $\mathcal{W}$, feasibility of QP-SM is not guaranteed anymore. If QP-SM is found to be still feasible by the check (see Fig. 1), standard mode is maintained; otherwise, the algorithm goes into recovery mode. In particular, if QP-STA is solvable, then timing is adapted to guarantee feasibility of QP-RM, which then generates a stable gait with modified footsteps.

At the end of the RM iteration, three outcomes are possible:

(1) The next QP-SM is feasible; therefore, standard mode is resumed.
(2) QP-SM is unfeasible, but QP-STA is feasible; therefore, the algorithm performs one more iteration in recovery mode.
(3) Neither QP-SM nor QP-STA are feasible; therefore, the proposed method cannot generate a stable gait in the current conditions. To handle this case, some sort of emergency stop procedure should be implemented.

11.3. Indirect disturbance compensation

The previous results, both in simulations and experiments, have shown that the proposed scheme is able to produce stable gaits in spite of the presence of disturbances. One may regard this positive result as an *indirect* disturbance compensation: in particular, as disturbance rejection when the original footstep plan is maintained (standard mode), and as disturbance attenuation when an adaptation is required (recovery mode). For example, in the case of constant disturbances (see LIP Simulations 3 and 5 and HRP-4 Dynamic Simulation 1) the robot counteracts the persistent lateral force by leaning against it, i.e., by displacing the CoM towards the perturbation.

An intuitive explanation of the leaning behavior can be given as follows. Consider a humanoid which must maintain balance in the presence of a constant acceleration w_x acting on the CoM. Steady-state equilibrium requires $\dot{x}_c = \ddot{x}_c = 0$, so that from the perturbed x-LIP equation (4) it follows that the CoM-ZMP displacement must be

$$x_c - x_z = \frac{w_x}{\eta^2}.$$

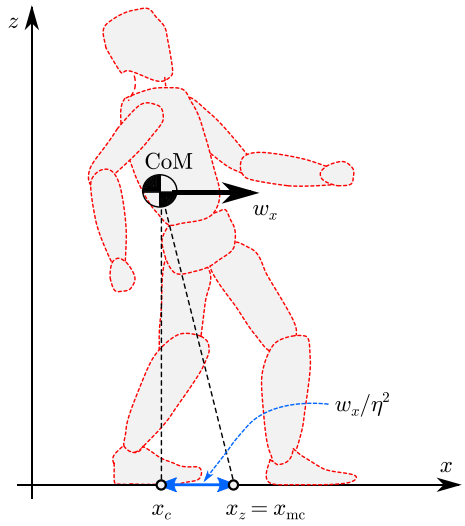


Fig. 23. Balancing against a constant disturbance w_x requires a steady-state COM-ZMP displacement which depends on w_x. In our scheme, which keeps the ZMP at the center of the moving constraint, this results in the CoM leaning with respect to the vertical against w_x.

In our scheme, this condition is recovered by using the optimal balancing condition ($x_z = x_{\mathrm{mc}}$, $\dot{x}_c = 0$) in the stability constraint (15), which provides

$$x_u = x_c + \frac{\dot{x}_c}{\eta} = x_c = x_{\mathrm{mc}} + \frac{w_x}{\eta^2},$$

i.e., the required CoM-ZMP displacement. In particular, since the ZMP is kept at the center of the moving constraint, this displacement will be realized by the CoM, ultimately producing the observed leaning behavior (see Fig. 23).

Although the above argument refers to the pure balancing case, it also remains valid when the robot is walking, the only difference being that x_{mc}, which is the reference for x_z, becomes a time-varying signal. It should also be noted that, since the causal version (23) of the stability constraint uses the disturbance estimate $\bar{w}_x$, the robot will actually lean against such estimate.

11.4. Setting the parameters of the expected disturbance set

In the RIS-MPC scheme, the expected disturbance set $\mathcal{W}$ is used to compute the ZMP restriction function so as to guarantee recursive feasibility (see Section 4.3). A precise characterization of $\mathcal{W}$ is available in many cases, but there are also situations in which the parameters that define $\mathcal{W}$ (namely, the midrange $w_{m,i}$ and the maximum deviation Δ_i of the disturbance, $i \in \{x, y\}$) must be conjectured based on heuristic considerations.

In particular, the designer might be tempted to increase Δ_i to enlarge the set of tolerated disturbances, i.e, disturbance for which recursive feasibility in standard mode is guaranteed. However, two facts should be considered:

- there is an upper bound on Δ_i, above which it not possible to find a suitable restriction function, see Eq. (36);
- drastic restriction functions may lead to a severe reduction of the feasibility region $\mathcal{F}_S$, which is undesirable for standard mode initialization and resumption.

In view of the above, a reasonable trade-off would be to define the expected disturbance set so as to include the most frequent disturbances up to a suitable limit, while leaving the recovery mode to take care of occasional large disturbances outside $\mathcal{W}$.

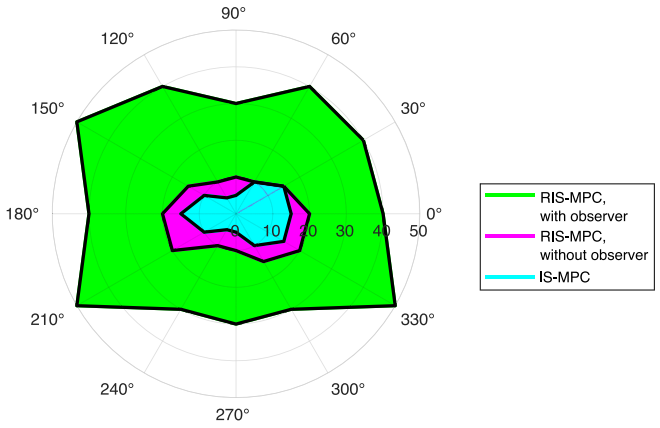


Fig. 24. The same campaign of simulations of Fig. 18, now also showing the performance of RIS-MPC when the observer is removed from the disturbance estimator.

11.5. The role of the observer in the disturbance estimator

Proposition 3 provides sufficient conditions for recursive feasibility of the RIS-MPC scheme. Analyzing its proof, it becomes evident that the specific observer used within the disturbance estimator does not matter, as only the conditions enforced by the saturator block are necessary to establish the result. Indeed, one may actually remove the observer altogether, using directly the midrange value of the disturbances $w_{m,x}$, $w_{m,y}$ as $\hat{w}_x$, $\hat{w}_{m,y}$, and still guarantee recursive feasibility for disturbances belonging to the expected set $\mathcal{W}$.

In the presence of disturbances that *exceed* the set $\mathcal{W}$, the conditions of Proposition 3 are invalidated, but QP-SM can still be feasible provided that (x_u^k, y_u^k) remains in $\mathcal{F}_S^k$. This is exactly where the observer block plays a beneficial role, as it increases the practical robustness of the scheme against such disturbances. This is confirmed by Fig. 24, which replicates the same results of Fig. 18 with the additional indication of what the performance of RIS-MPC would be if the observer was removed. The robustness improvement achieved thanks to the observer is substantial, and is due to the small estimation error which it guarantees (see the discussion in the last paragraph of Section 4.2).

The inclusion of an observer in the disturbance estimator has another advantage: it makes the controller more robust to *variations* of the disturbance. To support this claim, consider the performance index

$$C_{\mathrm{fea}}^k = \min_{(x,y)\in\partial\mathcal{F}_S^k} \sqrt{(x_u^k - x)^2 + (y_u^k - y)^2},$$

which represents the *feasibility clearance* at t_k, i.e., the distance between (x_u^k, y_u^k) and the boundary of the feasibility region[9] $\mathcal{F}_S^k$. The higher is C_{fea}^k, the larger is the intensity of the *additional* disturbance that can be tolerated in all directions. Simulation results show that the inclusion of the asymptotic observer (7)–(8) in the disturbance estimator displaces $\mathcal{F}_S^k$ in such a way that (x_u^k, y_u^k) remains closer to its center, thus resulting in larger values of C_{fea}^k. This is shown qualitatively in Fig. 25 and quantitatively in Fig. 26; in particular, the latter indicates that, after an initial transient, C_{fea}^k is always larger when the observer is used, as opposed to when the observer is disabled.

11.6. Choosing the parameters of the disturbance estimator

Since the disturbance estimator consists of both the asymptotic observer and the saturator (see Fig. 1), its parameters include the rapidity of the first (which essentially depends on the eigenvalues of

[9] We focus on the feasibility of the standard mode, but the following discussion can be repeated for the recovery mode.

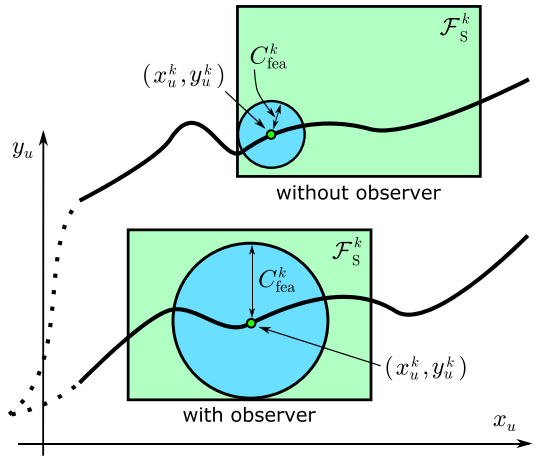


Fig. 25. Illustration of the advantage of increasing the feasibility clearance: the inclusion of the asymptotic observer in the disturbance estimator keeps the feasibility region $\mathcal{F}_S^k$ more centered around the current state, ultimately increasing the disturbance variation that can be tolerated in all directions.

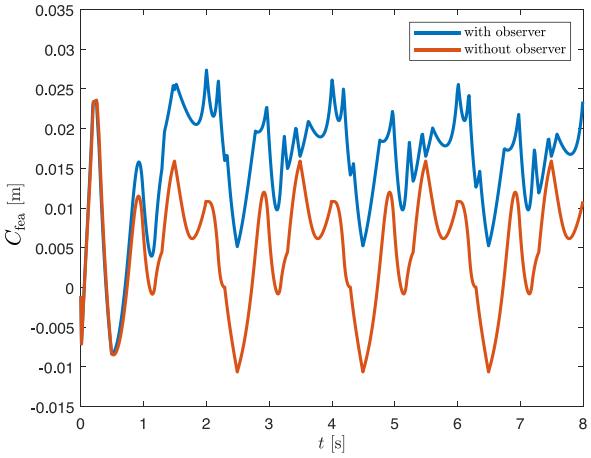


Fig. 26. The effect of the asymptotic observer on the feasibility clearance C_{fea} in a typical simulation.

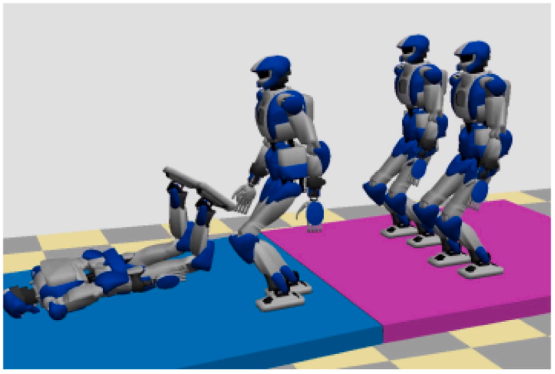


Fig. 27. Comparison between RIS-MPC (robot in the foreground) and RCL-MPC (robot in the background). While both robots are able to walk down the first part of the ramp, only RIS-MPC can handle the doubled slope (see the accompanying video).

the observation error dynamics) as well as the maximum estimate increments w_x^{diff}, w_y^{diff} used in the second (see Eq. (9b)).

The discussion of Section 11.5 shows that the robustness of RIS-MPC is improved if the estimation error converges quickly. In addition to requiring a fast observer, this also suggests to remove the saturator at runtime; however, doing so would invalidate the recursive feasibility proof. A softer alternative would be to increase the parameters of the saturator so as to moderate its effect; recursive feasibility would then be preserved, although for a reduced disturbance range. Ultimately, one may say that removing or moderating the saturator favors practical over guaranteed performance, so that the designer must identify a proper trade-off.

Finally, it should be noted that the RIS-MPC scheme does not require a specific design choice for the asymptotic observer. In fact, a different observer or a Kalman Filter [7,37] can be used in place of (7)–(8).

11.7. Comparison with the state of the art

We are now in a suitable position to provide a comparative discussion of the proposed RIS-MPC scheme with respect to some state-of-the-art methods for robust gait generation which were already briefly mentioned in the introduction.

The authors of [44] propose to use a measure of the external force to derive a modified dynamic balance condition to be used in an MPC-based algorithm. This scheme can tolerate persistent disturbances by generating CoM trajectories which exhibit a leaning behavior similar to that of our method. Footsteps are regarded as decision variables, implying a certain robustness to impulsive perturbations. While there are certain similarities with our approach, at least two major differences can be identified: no stability constraint is enforced (thus, recursive feasibility is not addressed), and no step timing modification is allowed.

The method proposed in [16] addresses robustness by using footstep adaptation and projecting the state of the robot into an appropriate viability kernel in order to prevent divergent CoM-ZMP trajectories. If the measured state of the robot is non-viable (i.e., it invariably leads to failed states), it is projected inside a region from which at least one feasible trajectory exists. With this approach, feasibility of the prediction is always guaranteed, but no proof of robust stability can be given for the closed-loop system. Furthermore, the projection strategy is activated once feasibility is lost, meaning that, differently from RIS-MPC, the method is not proactive toward *maintaining* robustness.

The most interesting comparison can be made between our method and the RCL-MPC approach [5], in which the concept of robust positive invariance is used to design a gait generation method which can work under perturbations. A DCM feedback works in conjunction with an MPC controller, and safety margins are introduced in the ZMP constraints to guarantee that the CoM tracking error remains within given bounds. The use of safety margins is similar to the ZMP constraint restriction used in RIS-MPC, which however is aimed at ensuring recursive feasibility of the scheme, and therefore stability of the gait.

The authors of [5] provide an estimate of the maximum constant disturbance tolerated by RCL-MPC, which using our notation translates to

$$\Delta^{\max} = \eta^2 \frac{d_z}{4} e^{-\eta\delta}. \tag{46}$$

For RIS-MPC, the upper bound on disturbances for which recursive feasibility is guaranteed is given by (36). Setting $w_i^{\text{diff}} = 0$ (i.e., using the midrange value as an estimate of the disturbance) and noting that the second term is typically negligible (as $T_p \gg \delta$), only the first term remains, which coincides with $\Delta^{\max}$.

Still, the performance of the two schemes can be quite different. Thanks to the indirect disturbance compensation mechanism, RIS-MPC is able to keep the ZMP close to the center of the ZMP constraint as long as the disturbance is estimated with sufficient accuracy. The effect of the disturbance is then seen on the CoM trajectory, which displays the leaning behavior discussed before. Conversely, RCL-MPC does not perform disturbance compensation and therefore produces ZMP trajectories that are displaced in the direction of the disturbance; clearly, this

can be a problem because there is a limit to this displacement, i.e., the boundary of the ZMP constraint itself.

To confirm the above analysis, we ran a comparative simulation (see Fig. 27) in which two HRP-4 robots, one controlled by RIS-MPC and the other by RCL-MPC, are commanded to walk down a ramp with variable slope. Both robots tolerate the initial slope of 2.5°, but only RIS-MPC can handle the second part of the ramp in which the slope is doubled.

Finally, it should be noted that RCL-MPC does not adapt timing or footsteps, and therefore only limited robustness can be expected in the presence of impulsive perturbations.

12. Conclusions

We have presented RIS-MPC, a framework based on predictive control for generating humanoid gaits in the presence of perturbations, both persistent and impulsive. RIS-MPC employs two modes of operation, respectively called standard mode (SM) and recovery mode (RM).

In SM, an optimal control action is computed under a stability constraint based on an estimate of the disturbance and a suitably restricted ZMP constraint. It was shown that recursive feasibility of SM is ensured if the perturbations are bounded in a known set.

The RM is activated when QP-SM becomes unfeasible due to a perturbation exceeding the known set. To recover feasibility, no restriction is used on the ZMP constraints and the footstep positions are adapted. Moreover, a preliminary Step Timing Adaptation (STA) procedure is used to make sure that the current state will be contained in the feasibility region of the RM.

The proposed approach was extensively validated by simulations (ranging from MATLAB simulations on the LIP model to dynamic DART simulations on the HRP-4 humanoid) as well as experiments on the small-size humanoids NAO and OP3. Overall, it was shown that the RIS-MPC method for gait generation is effective and robust to different kinds of perturbations.

We point out that RIS-MPC can be also implemented on torque-controlled humanoids. To this end, the reference CoM trajectories — which in this paper were directly sent to the kinematic controller — must be augmented with references for the ground reaction forces at the foot (feet) in contact. The resultant of these ground reaction forces can be computed from the Newton equation on the full robot using the estimates $(\tilde{w}_x, \tilde{w}_y)$ to account for the presence of disturbances. During double support, the resultant ground reaction force must be distributed between the two feet using a suitable criterion, e.g., by specifying a temporal transition law that defines the share of the force applied on each foot as a function of the time passed since the beginning of double support. Note that these forces automatically satisfy the unilaterality condition, which is equivalent to the ZMP condition that has been enforced at the MPC level. Moreover, one might consider additional constraints (e.g., to ensure sufficient friction) by formulating a simple constrained optimization problem. Once the force references are computed, whole-body control can be achieved using standard inverse dynamics techniques.

Another interesting development of the proposed scheme could be the on-line adaptation of the disturbance bounds: since our disturbance observer does not need to know the bounds prior to the saturator, the (unsaturated) maximum value reached by the estimated disturbance can be stored and used to adapt the bounds in the restriction function and saturator stage. A further possible extension that may be investigated in the future is the inclusion of the full robot kinematics in the optimization process, as proposed in [45].

CRediT authorship contribution statement

Nicola Scianca: Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Filippo M. Smaldone:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Software, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Leonardo Lanari:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Methodology, Investigation, Formal analysis, Data curation, Conceptualization. **Giuseppe Oriolo:** Writing – review & editing, Writing – original draft, Visualization, Validation, Supervision, Methodology, Investigation, Formal analysis, Data curation, Conceptualization.

Declaration of competing interest

The authors declare the following financial interests/personal relationships which may be considered as potential competing interests: Nicola Scianca reports financial support was provided by PNRR MUR project PE0000013-FAIR. If there are other authors, they declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A

For the reader's convenience, we collect here the proofs of the propositions in the paper.

Proof of Proposition 1. It is a simple adaptation of the proof of Prop. 1 in [46] to the perturbed x-LIP. In particular, simply replace x_z in that proof with $u = x_z + w_x/\eta$, and consider that u satisfies the requirement of being linearly bounded, because by hypothesis both x_z and w_x have this property. In particular, x_z is linearly bounded because $\dot{x}_z$ is assumed to be piecewise-constant, while w_x belongs to the bounded set (3). ■

Proof of Proposition 2. Necessity can be demonstrated by multiplying each term in the restricted ZMP constraint (11) by $\eta\, e^{-\eta(t-t_k)}$, integrating from t_k to t_{k+C}, adding $\tilde{c}_x^k$ and subtracting $\tilde{w}_x^k/\eta^2$ from each side of the inequality. The middle term of the resulting three-side inequality can be shown to coincide with x_u^k, thanks to (23), while the left and right terms are $x_{u,\mathrm{S}}^{k,m}$ and $x_{u,\mathrm{S}}^{k,M}$, respectively, as given in the thesis.

To prove sufficiency, assume that $x_{u,\mathrm{S}}^{k,m} \le x_u^k \le x_{u,\mathrm{S}}^{k,M}$, so that one can write

$$x_u^k = a\, x_{u,\mathrm{S}}^{k,m} + (1-a)\, x_{u,\mathrm{S}}^{k,M}, \quad a \in [0,1]. \tag{47}$$

Then, the ZMP trajectory

$$x_z(t) = a(x_z^m(t) + R_x(t-t_k)) + (1-a)(x_z^M(t) - R_x(t-t_k)) \tag{48}$$

is feasible, because it satisfies the restricted ZMP constraint (by construction) as well as the robust stability constraint. The latter fact can be easily verified by plugging (48) in (23), and using (47) together with $\tilde{c}_x^k = a\tilde{c}_x^k + (1-a)\tilde{c}_x^k$ and $\tilde{w}_x^k = a\tilde{w}_x^k + (1-a)\tilde{w}_x^k$.

The proof for y is similar. ■

Proof of Proposition 3. Assume that QP-SM is feasible at t_k. Our objective is to show that it will also be feasible at t_{k+1}.

As before, let us focus on motion along the x axis only. Feasibility at t_k implies that there exists a control sequence $\dot{x}_z^k, \dots, \dot{x}_z^{k+C-1}$ such that both the restricted ZMP constraint (11) and the robust stability constraint (24) are satisfied; the latter implies that also (23) holds. By integrating the dynamics (14) of x_u, we get

$$x_u^{k+1} = e^{\eta\delta} x_u^k - \int_{t_k}^{t_{k+1}} e^{\eta(t_{k+1}-\tau)} \left(\eta x_z(\tau) - w_x(\tau)/\eta\right) d\tau.$$

Plugging (23) into the above equation, and using (11), (20) and (22), we obtain

$$x_u^{k+1} \leq \eta \int_{t_{k+1}}^{t_{k+C}} e^{-\eta(\tau-t_{k+1})}(x_z^M(\tau) - R_x(\tau - t_k))d\tau + \eta \int_{t_{k+C}}^{\infty} e^{-\eta(\tau-t_{k+1})}\tilde{x}_z(\tau)d\tau + \frac{1}{\eta}\int_{t_k}^{t_{k+1}} e^{-\eta(\tau-t_{k+1})}w_x(\tau)d\tau - \frac{e^{\eta\delta}}{\eta^2}\tilde{w}_x^k. \quad (49)$$

According to Proposition 2, feasibility of QP-SM at time t_{k+1} requires $x_{u,\mathrm{S}}^{k+1,m} \leq x_u^{k+1} \leq x_{u,\mathrm{S}}^{k+1,M}$. The right side inequality can be written as

$$x_u^{k+1} \leq \eta \int_{t_{k+1}}^{t_{k+C+1}} e^{-\eta(\tau-t_{k+1})}(x_z^M(\tau) - R_x(\tau - t_{k+1}))d\tau + \eta \int_{t_{k+C+1}}^{\infty} e^{-\eta(\tau-t_{k+1})}\tilde{x}'_z(\tau)d\tau - \frac{1}{\eta^2}\tilde{w}_x^{k+1}, \quad (50)$$

where $\tilde{x}'_z(\tau)$ is the ZMP tail used in the robust stability constraint at t_{k+1}.

To guarantee that QP-SM is feasible at t_{k+1}, we impose that the right bound of (49) is not larger than the right bound of (50). Rearranging and simplifying leads to the following inequality

$$\beta_x \geq \phi_x^k + \psi_x^k, \quad (51)$$

where β_x is obtained by setting $i = x$ in (33), and

$$\phi_x^k = \eta \int_{t_{k+C+1}}^{\infty} e^{-\eta(\tau-t_{k+1})}(\tilde{x}_z(\tau) - \tilde{x}'_z(\tau))d\tau - (e^{\eta\delta} - 1)\frac{e^{-\eta T_c} d_{z,x}}{2} \quad (52)$$

$$\psi_x^k = \frac{\tilde{w}_x^{k+1} - \tilde{w}_x^k}{\eta^2} + \frac{1}{\eta}\int_{t_k}^{t_{k+1}} e^{-\eta(\tau-t_{k+1})}(w_x(\tau) - \tilde{w}_x^k)d\tau. \quad (53)$$

In computing ϕ_x^k, we used the fact that $x_z^M - \tilde{x}_z = d_{z,x}/2$ thanks to (19).

In Eq. (51), β_x is independent from k, but both terms in the right-hand side depend on k. To derive a sufficient condition for recursive feasibility, we must replace each of them with an upper bound over k.

In particular, we have

$$\phi_x^k \leq (e^{\eta\delta} - 1)\left(\frac{e^{-\eta T_p}}{\eta} v_{z,x}^{\max} - \frac{e^{-\eta T_c}}{2} d_{z,x}\right) = \phi_x,$$

where we have considered that the first term in (52) (the *tail mismatch* due to T_p being finite) can be bounded as

$$\eta \int_{t_{k+P+1}}^{\infty} e^{-\eta(\tau-t_{k+1})}(\tilde{x}_z(\tau) - \tilde{x}'_z(\tau))d\tau \leq e^{-\eta T_p}\frac{e^{\eta\delta} - 1}{\eta} v_{z,x}^{\max}.$$

being $v_{z,x}^{\max}$ is the maximum x-velocity of the ZMP in the anticipative tail

As for ψ_x^k, which is given by (53), we can write

$$\psi_x^k \leq \frac{w_x^{\mathrm{diff}}}{\eta^2} + \frac{2\Delta_x(e^{\eta\delta} - 1)}{\eta^2} = \psi_x,$$

having used the constraints enforced by the saturator stage of the disturbance estimator and the fact that $w_x \in \mathcal{W}_x$.

Wrapping up, to guarantee $x_u^{k+1} \leq x_{u,\mathrm{S}}^{k+1,M}$, for all k, we must replace ϕ_x^k and ψ_x^k in (51) with their upper bounds ϕ_x and ψ_x; this yields condition (31) of the thesis, with β_x, ϕ_x and ψ_x respectively given by (33), (34) and (35), for $i = x$. One may easily verify that the same condition also guarantees the lower bound for feasibility at t_{k+1} (i.e., $x_{u,\mathrm{S}}^{k+1,m} \leq x_u^{k+1}$), and therefore it is a sufficient condition for recursive feasibility in x.

The above arguments can be repeated for the motion along the y axis, leading to the sufficient condition (31) for recursive feasibility in y. This completes the proof. ■

Proof of Proposition 4. The result can be easily proven by using the same reasoning of [30]. In particular, one first computes the actual feasibility region $\mathcal{F}_{\mathrm{R}}^k$ of QP-RM, obtaining a result similar to [30, Prop. 1], the only difference being that the robust stability constraints (24) and (28) are used in place of their original counterpart. The second step is to follow the procedure in [30, Sect. IV-B] to derive the conservative estimate $\mathcal{F}_{\mathrm{R,est}}^k$, whose bounds now include additive terms in $\tilde{w}_x^k$, $\tilde{w}_y^k$ as given in the thesis. ■

Appendix B. Supplementary data

Supplementary material related to this article can be found online at https://doi.org/10.1016/j.robot.2025.104957.

Data availability

Data will be made available on request.

References

[1] P.-B. Wieber, Trajectory free linear model predictive control for stable walking in the presence of strong perturbations, in: 6th IEEE-RAS Int. Conf. on Humanoid Robots, 2006, pp. 137–142.

[2] L. Chisci, J.A. Rossiter, G. Zappa, Systems with persistent disturbances: predictive control with restricted constraints, Automatica 37 (7) (2001) 1019–1028.

[3] E.C. Kerrigan, Robust Constraint Satisfaction: Invariant Sets and Predictive Control (Ph.D. thesis), University of Cambridge, 2000.

[4] D. Limon, I. Alvarado, T. Alamo, Robust tube-based MPC for tracking of constrained linear systems with additive disturbances, J. Process Control 20 (3) (2010) 248–260.

[5] N. Villa, P.-B. Wieber, Model predictive control of biped walking with bounded uncertainties, in: 17th IEEE-RAS Int. Conf. on Humanoid Robots, 2017, pp. 836–841.

[6] A. Gazar, M. Khadiv, A. Del Prete, L. Righetti, Stochastic and robust MPC for bipedal locomotion: A comparative study on robustness and performance, in: 20th IEEE-RAS Int. Conf. on Humanoid Robots, 2021, pp. 61–68.

[7] B.J. Stephens, State estimation for force-controlled humanoid balance using simple models in the presence of modeling error, in: 2011 IEEE Int. Conf. on Robotics and Automation, 2011, pp. 3994–3999.

[8] K. Kaneko, F. Kanehiro, M. Morisawa, E. Yoshida, J.-P. Laumond, Disturbance observer that estimates external force acting on humanoid robots, in: 12th IEEE Int. Workshop on Advanced Motion Control, 2012, pp. 1–6.

[9] L. Hawley, W. Suleiman, External force observer for medium-sized humanoid robots, in: 2016 IEEE-RAS Int. Conf. on Humanoid Robots, 2016, pp. 366–371.

[10] J. Li, Z. Yuan, S. Dong, J. Zhang, X. Sang, External force observer aided push recovery for torque-controlled biped robots, Auton. Robots 46 (5) (2022) 553–568.

[11] S. Czarnetzki, S. Kerner, O. Urbann, Observer-based dynamic walking control for biped robots, Robot. Auton. Syst. 57 (8) (2009) 839–845.

[12] R.J. Griffin, A. Leonessa, A. Asbeck, Disturbance compensation and step optimization for push recovery, in: 2016 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2016, pp. 5385–5390.

[13] J. Englsberger, G. Mesesan, C. Ott, Smooth trajectory generation and push-recovery based on divergent component of motion, in: 2017 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2017, pp. 4560–4567.

[14] D.J. Agravante, A. Cherubini, A. Sherikov, P.-B. Wieber, A. Kheddar, Human-humanoid collaborative carrying, IEEE Trans. Robot. 35 (4) (2019) 833–846.

[15] A. Herdt, H. Diedam, P.-B. Wieber, D. Dimitrov, K. Mombaur, M. Diehl, Online walking motion generation with automatic footstep placement, Adv. Robot. 24 (5–6) (2010) 719–737.

[16] M.H. Yeganegi, M. Khadiv, A. Del Prete, S.A.A. Moosavian, L. Righetti, Robust walking based on MPC with viability guarantees, IEEE Trans. Robot. 38 (4) (2021) 2389–2404.

[17] P. Kryczka, P. Kormushev, N. Tsagarakis, D.G. Caldwell, Online regeneration of bipedal walking gait optimizing footstep placement and timing, in: 2015 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2015, pp. 3352–3357.

[18] N. Bohórquez, P. Wieber, Adaptive step duration in biped walking: A robust approach to nonlinear constraints, in: 17th IEEE-RAS Int. Conf. on Humanoid Robots, 2017, pp. 724–729.

[19] S. Caron, Q. Pham, When to make a step? Tackling the timing problem in multi-contact locomotion by TOPP-MPC, in: 17th IEEE-RAS Int. Conf. on Humanoid Robots, 2017, pp. 522–528.

[20] Y. Kojio, Y. Omori, K. Kojima, F. Sugai, Y. Kakiuchi, K. Okada, M. Inaba, Footstep modification including step time and angular momentum under disturbances on sparse footholds, IEEE Robot. Autom. Lett. 5 (3) (2020) 4907–4914.

[21] A. Ibanez, P. Bidaud, V. Padois, Emergence of humanoid walking behaviors from mixed-integer model predictive control, in: 2014 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2014, pp. 4014–4021.

[22] M.R.O.A. Maximo, C.H.C. Ribeiro, R.J.M. Afonso, Mixed-integer programming for automatic walking step duration, in: 2016 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2016, pp. 5399–5404.
[23] M. Shafiee, G. Romualdi, S. Dafarra, F.J.A. Chavez, D. Pucci, Online DCM trajectory generation for push recovery of torque-controlled humanoid robots, in: 19th IEEE-RAS Int. Conf. on Humanoid Robots, 2019, pp. 671–678.
[24] H. Jeong, I. Lee, J. Oh, K.K. Lee, J.-H. Oh, A robust walking controller based on online optimization of ankle, hip, and stepping strategies, IEEE Trans. Robot. 35 (6) (2019) 1367–1386.
[25] R.J. Griffin, G. Wiedebach, S. Bertrand, A. Leonessa, J. Pratt, Walking stabilization using step timing and location adjustment on the humanoid robot, Atlas, in: 2017 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2017, pp. 667–673.
[26] N. Scianca, M. Cognetti, D. De Simone, L. Lanari, G. Oriolo, Intrinsically stable MPC for humanoid gait generation, in: 16th IEEE-RAS Int. Conf. on Humanoid Robots, 2016, pp. 101–108.
[27] N. Scianca, D. De Simone, L. Lanari, G. Oriolo, MPC for humanoid gait generation: Stability and feasibility, IEEE Trans. Robot. 36 (4) (2020) 1171–1188.
[28] F.M. Smaldone, N. Scianca, V. Modugno, L. Lanari, G. Oriolo, Gait generation using intrinsically stable MPC in the presence of persistent disturbances, in: 19th IEEE-RAS Int. Conf. on Humanoid Robots, 2019, pp. 682–687.
[29] F.M. Smaldone, N. Scianca, V. Modugno, L. Lanari, G. Oriolo, ZMP constraint restriction for robust gait generation in humanoids, in: 2020 IEEE Int. Conf. on Robotics and Automation, 2020, pp. 8739–8745.
[30] F. Smaldone, N. Scianca, L. Lanari, G. Oriolo, Feasibility-driven step timing adaptation for robust MPC-based gait generation in humanoids, IEEE Robot. Autom. Lett. 6 (2) (2021) 1582–1589.
[31] J. Chestnutt, M. Lau, G. Cheung, J. Kuffner, J. Hodgins, T. Kanade, Footstep planning for the honda asimo humanoid, in: 2005 IEEE Int. Conf. on Robotics and Automation, 2005, pp. 629–634.
[32] A. Hornung, A. Dornbush, M. Likhachev, M. Bennewitz, Anytime search-based footstep planning with suboptimality bounds, in: 12th IEEE-RAS Int. Conf. on Humanoid Robots, 2012, pp. 674–679.
[33] P. Ferrari, N. Scianca, L. Lanari, G. Oriolo, An integrated motion planner/controller for humanoid robots on uneven ground, in: 18th European Control Conf., 2019, pp. 1598–1603.
[34] M. Cipriano, P. Ferrari, N. Scianca, L. Lanari, G. Oriolo, Humanoid motion generation in a world of stairs, Robot. Auton. Syst. 168 (2023) 104495.
[35] S. Kajita, H. Hirukawa, K. Harada, K. Yokoi, Introduction to Humanoid Robotics, Springer Publishing Company Inc., 2014.
[36] A. Tanguy, D. De Simone, A.I. Comport, G. Oriolo, A. Kheddar, Closed-loop MPC with dense visual SLAM - stability through reactive stepping, in: 2019 IEEE Int. Conf. on Robotics and Automation, 2019, pp. 1397–1403.
[37] L. Hawley, R. Rahem, W. Suleiman, Kalman filter based observer for an external force applied to medium-sized humanoid robots, in: 2018 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2018, pp. 1204–1211.
[38] B. Friedland, Control System Design: An Introduction to State-Space Methods, Courier Corporation, 2012.
[39] J. Pratt, J. Carff, S. Drakunov, A. Goswami, Capture point: A step toward humanoid push recovery, in: 6th IEEE-RAS Int. Conf. on Humanoid Robots, 2006, pp. 200–207.
[40] T. Takenaka, T. Matsumoto, T. Yoshiike, Real time motion generation and control for biped robot - 1st report: Walking gait pattern generation, in: 2009 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2009, pp. 1084–1091.
[41] G. Frison, M. Diehl, HPIPM: a high-performance quadratic programming framework for model predictive control, IFAC-PapersOnLine 53 (2) (2020) 6563–6569.
[42] N. Scianca, V. Modugno, L. Lanari, G. Oriolo, Gait generation via Intrinsically Stable MPC for a multi-mass humanoid model, in: 17th IEEE-RAS Int. Conf. on Humanoid Robots, 2017, pp. 547–552.
[43] L. Hawley, W. Suleiman, Control framework for cooperative object transportation by two humanoid robots, Robot. Auton. Syst. 115 (2019) 1–16.
[44] D.J. Agravante, A. Sherikov, P.-B. Wieber, A. Cherubini, A. Kheddar, Walking pattern generators designed for physical collaboration, in: 2016 IEEE Int. Conf. on Robotics and Automation, 2016, pp. 1573–1578.
[45] T. Belvedere, N. Scianca, L. Lanari, G. Oriolo, Joint-level IS-MPC: A whole-body MPC with centroidal feasibility for humanoid locomotion, in: 2024 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems, 2024, pp. 5399–5404.
[46] F. Smaldone, N. Scianca, L. Lanari, G. Oriolo, From walking to running: 3D humanoid gait generation via MPC, Front. Robot. AI 9 (2022).

Nicola Scianca received his Ph.D. in control engineering in 2022 from Sapienza University of Rome, Italy. He is currently a Researcher at the Department of Computer, Control and Management Engineering (DIAG) of the same university. He was a visiting scientist at the Model Predictive Control Laboratory of the University of California at Berkeley (CA, USA), the CNRS-AIST Joint Robotics Laboratory (Tsukuba, Japan) and the Inria Centre at Université de Lorraine (Nancy, France). His research interests focus on model predictive control for humanoid robots.

Filippo M. Smaldone received the M.Sc. and the Ph.D. in control engineering in 2019 and 2023, respectively, from Sapienza University of Rome, Italy. He is currently an RD engineer working in the aerospace industry. His research interests are in dynamic and robust locomotion for humanoid robots using Model Predictive Control and in hybrid inertial navigation.

Leonardo Lanari received his Ph.D. in control engineering in 1992 from Sapienza University of Rome, Italy. He is currently with the Department of Computer, Control and Management Engineering (DIAG) of the same university as an Associate Professor of Automatic Control. His research interests are in the general area of planning and control of robotic systems, with an emphasis on humanoid control and robots with elastic joints and links.

Giuseppe Oriolo received his Ph.D. in control engineering in 1992 from Sapienza University of Rome, Italy. He is currently with the Department of Computer, Control and Management Engineering (DIAG) of the same university, where he is a Full Professor of Automatic Control and Robotics and the director of the DIAG Robotics Lab. His research interests are in the general area of planning and control of robotic systems. Prof. Oriolo has been Associate Editor of the IEEE Transactions on Robotics and Automation from 2001 to 2005 and Editor of the IEEE Transactions on Robotics from 2009 to 2013. He is a Fellow of the IEEE.