Feasibility-Driven Step Timing Adaptation for Robust MPC-Based Gait Generation in Humanoids

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

***Abstract*—The feasibility region of a Model Predictive Control (MPC) algorithm is the subset of the state space in which the constrained optimization problem to be solved is feasible. In our recent Intrinsically Stable MPC (IS-MPC) method for humanoid gait generation, feasibility means being able to satisfy the dynamic balance condition, the kinematic constraints on footsteps as well as an explicit stability condition. Here, we exploit the feasibility concept to build a step timing adapter that, at each control cycle, modifies the duration of the current step whenever a feasibility loss is imminent due, e.g., to an external perturbation. The proposed approach allows the IS-MPC algorithm to maintain its linearity and adds a negligible computational burden to the overall scheme. Simulations and experimental results where the robot is pushed while walking showcase the performance of the proposed approach.**

***Index Terms*—Humanoids and Bipedal Locomotion, Robust/Adaptive Control, Optimization and Optimal Control**

I. Introduction

ROBUST locomotion is a central issue of humanoid robotics. Humans mostly use regular gaits with equal steps, but can change step length and timing as necessary. Current algorithms for gait generation are limited in this respect, as they must deal with the complexity of the walking dynamics while guaranteeing real-time performance: often, step length can be adapted while step timings are predefined.

To avoid falling, balance must be guaranteed at all times. This can be realized by keeping the Zero Moment Point (ZMP) within the support polygon. To achieve real-time control, a popular approach is to represent humanoid dynamics by the Linear Inverted Pendulum (LIP) model, characterized by an unstable mode often referred to as Divergent Component of Motion (DCM) [1]. The linearity of the LIP allows tracking predefined ZMP trajectories with a preview-based approach [2], or encoding the ZMP balance requirement through constraints, leading to an MPC formulation [3].

While early MPC algorithms only generated a suitable Center of Mass (CoM) trajectory for a given set of footsteps, later methods [4] were also able to plan footsteps while maintaining the linearity of the constraints. When it comes to adapting step timing, however, the extension in an MPC framework is not trivial since it would invariably involve either nonlinear constraints or discrete optimization.

A nonlinear MPC is presented in [5] for adapting step timings via time scaling at each iteration. The MPC step of [6] works in two stages: CoM/swing foot paths are generated by interpolation and then a suitable timing is computed by optimization. Nonlinear optimization is also used in [7] for generating footsteps and timings in response to perturbations. An alternative [8], [9] is to use boolean variables in the MPC, leading to Mixed-Integer Quadratic Programming (MIQP). Both nonlinear and MIQP-based MPC are computationally heavy and cannot guarantee convergence of the optimizer.

As for non-MPC approaches, a QP problem is formulated in [10] for a point-foot LIP in terms of the DCM offset w.r.t. the center of the current support foot, in order to adapt both the position and timing of the next footstep. Other works such as [11] and [12] have developed similar ideas, while the DCM-based algorithm in [13] speeds up the step when the push occurs in the direction of the motion.

In this paper, we build upon our Intrinsically Stable MPC (IS-MPC) scheme, which guarantees recursive feasibility and stable gait generation thanks to an explicit stability constraint [14]. That result hinged on the study of the feasibility region of IS-MPC for the case of given footsteps. Here, we first extend the feasibility analysis of IS-MPC to the case of automatic footstep placement. Based on this, a Step Timing Adaptation (STA) module is designed, which can modify the current step duration to counteract an imminent loss of feasibility due to a perturbation. Since the feasibility region bounds are approximately linear in a suitable encoding of the current step duration, STA can be efficiently formulated and solved as a QP with a single variable. At every sampling instant, the STA and IS-MPC modules are run sequentially and the added computational load is negligible.

The paper is organized as follows. In Sect. II we give an outline of the proposed approach, while in Sect. III we review IS-MPC for gait generation. In Sect. IV we analyze the feasibility region for the case of automatic footstep placement, compute a linear conservative estimate of this region, and use it to perform step timing adaptation. Section V presents MATLAB simulations for the LIP, including a comparison with an STA technique based on the DCM offset. Finally, Sects. VI and VII show dynamic simulations in DART for an HRP-4 humanoid and experimental results on a NAO.

II. The Proposed Approach

Consider the problem of robust gait generation for a humanoid in the presence of perturbations, e.g., a push. We assume that the gait is generated in real time by MPC: in particular, we adopt the IS-MPC framework [14], which generates both CoM and ZMP trajectories so as to guarantee dynamic balance and internal stability. At the k-th time instant t_k, IS-MPC solves a constrained QP problem over a control

Manuscript received: October 15, 2020; Revised January 5, 2021; Accepted February 4, 2021. This letter was recommended for publication by Editor A. Kheddar upon evaluation of the Associate Editor and Reviewers' comments.

The authors are with the Dipartimento di Ingegneria Informatica, Automatica e Gestionale, Sapienza Università di Roma, via Ariosto 25, 00185 Roma, Italy. E-mail: {lastname}@diag.uniroma1.it.

Digital Object Identifier (DOI): see top of this page.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/LRA.2021.3059627, IEEE Robotics and Automation Letters

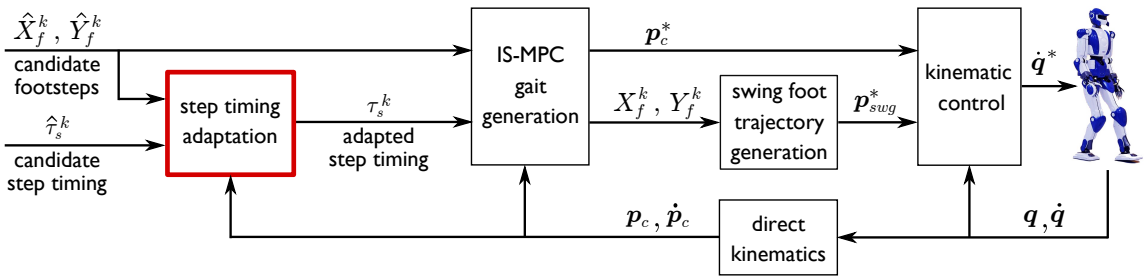

Fig. 1. A scheme of the proposed approach, with the step timing adaptation module acting before IS-MPC gait generation.

horizon, which is guaranteed to be solvable provided that the state of the robot belongs to a certain feasibility region $\mathcal{F}^k$. As a result of a perturbation, the state may be pushed outside such region, ultimately leading to a loss of balance. Since the feasibility region depends among other things on the duration of the steps, the basic idea of this paper is to adapt these durations in order to enlarge $\mathcal{F}^k$ as much as needed to absorb the effect of perturbations.

An overview of the proposed scheme is shown in Fig. 1. At each t_k, the input[1] consists of a sequence of F candidate footsteps in the x-y plane with the associated timing:

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

Here, $(\hat{x}_f^j, \hat{y}_f^j)$, $j = 1, \dots, F$, is the position of the j-th candidate footstep and $\hat{t}_s^j$ is the corresponding timestamp, indicating the time instant at which the (swinging) foot should land on $(\hat{x}_f^j, \hat{y}_f^j)$, i.e., the beginning of double support. The difference $\hat{T}_s^j = \hat{t}_s^{j+1} - \hat{t}_s^j$ is the duration of the j-th step, divided according to a fixed ratio in double support and single support, of duration $\hat{T}_{ds}^j$ and $\hat{T}_{ss}^j$, respectively.

In general, the MPC control horizon T_c will contain the remaining part of the *current* step (from t_k to $\hat{t}_s^1$); the first, second, ..., F-th subsequent steps; and the initial part of a last step (from $\hat{t}_s^F$ to t_{k+C}); see Fig. 2. Since the duration of steps is variable, the number F of footsteps falling within the current MPC control horizon T_c may change with k. In the following, we will need the last and the penultimate footsteps before t_k, denoted by (x_f^0, y_f^0) and (x_f^{-1}, y_f^{-1}), respectively (these are not candidates as they have already been realized); the corresponding timestamps are t_s^0 and t_s^{-1}.

All footsteps have the same orientation with respect to the x axis, chosen to be zero without loss of generality. This simplifying assumption allows to decouple the gait generation problem along the x (sagittal) and y (coronal) axes, and facilitates the computation of the feasibility region of IS-MPC. However, the proposed method can be extended in principle to the case of footsteps with variable orientation.

While the candidate footstep positions are directly fed to IS-MPC gait generation, the candidate timing goes through an adaptation module. If the current CoM of the robot is such that the next QP problem will be solvable, no modification is necessary; whereas if we are outside the feasibility region the timing will be changed so as to maintain feasibility. The timing after adaptation is denoted as $\tau_s^k = (t_s^1, \dots, t_s^F)$.

The actual inputs of IS-MPC gait generation will be the candidate footsteps and the adapted step timing (see Fig. 1).

[1]This input data typically comes from a footstep planner, which is not discussed in this paper. See [14] for a simple planner for flat ground that can be used in conjunction with IS-MPC, and [15] for a 3D planner.

III. IS-MPC

Before presenting the design of the step timing adaptation module, let us review the basics of gait generation based on IS-MPC (see [14] for details). This algorithm 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., $x(t_k) = x^k$. The MPC control horizon T_c covers C sampling intervals, i.e., $T_c = C\,\delta$.

Below we only consider gait generation for the x component. Every equation will have an identical counterpart for y component, except when explicitly stated.

IS-MPC uses a simplified model of the humanoid to characterize the relationship between the CoM and the ZMP, whose positions are respectively denoted by $\boldsymbol{p}_c = (x_c, y_c, z_c)$ and $\boldsymbol{p}_z = (x_z, y_z, 0)$. Assuming flat horizontal ground, constant CoM height and constant angular momentum around the CoM leads to the well-known LIP equation

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

where $\eta = \sqrt{g/\bar{z}_c}$, being $\bar{z}_c$ the constant CoM height. The prediction model in IS-MPC is a dynamically extended LIP

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

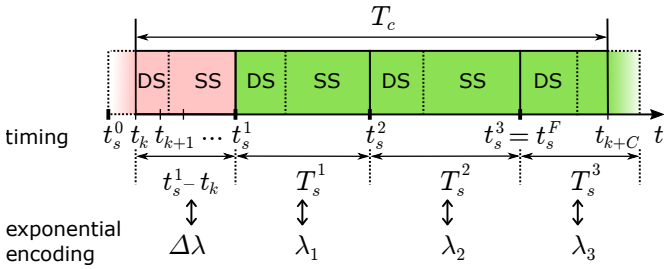

Fig. 2. The timing within the MPC control horizon T_c. Each step duration T_s^j is divided into a double support (DS) and a single support (SS) phase. The timestamp t_s^j associated to the j-th footstep indicates the start of the double support phase. Also shown are symbols which will be later used to exponentially encode the duration. Hats ($\hat{}$) are omitted but the picture is valid both for the candidate and the adapted step timing.

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 $t \in [t_k, t_{k+1})$.

The first constraint in IS-MPC is that the ZMP should lie inside the support polygon at all times. To express this as a linear constraint of the footsteps throughout the j-th step, we slide the footstep region during double support from the j-th to the $(j+1)$-th footstep. This allows to write the ZMP constraint as follows

$$\left| x_z - x_f^{j-1} - (x_f^j - x_f^{j-1})\sigma(t, t_s^j, t_s^j + T_{ds}^j) \right| \leq \frac{1}{2} d_{z,x}, \quad (1)$$

for $t \in [t_s^j, t_s^{j+1})$, having used for compactness the piecewise-linear sigmoidal function

$$\sigma(t, t_i, t_f) = \frac{1}{t_f - t_i} \left(\rho(t - t_i) - \rho(t - t_f) \right), \quad (2)$$

with $\rho(t) = t\, \delta_{-1}(t)$ the unit ramp starting at 0.

The second constraint ensures that footsteps are placed so as to be kinematically realizable by the robot. This means enforcing a maximum step length along the x axis

$$\left| x_f^j - x_f^{j-1} \right| \leq \frac{d_{a,x}}{2}, \quad (3)$$

with $d_{a,x}$ the size of the admissible region along x, see Fig. 3.

The last constraint in IS-MPC is needed to guarantee that the CoM remains bounded with respect to the ZMP. In fact, definition of the new coordinate[2] $x_u = x_c + \dot{x}_c/\eta$ highlights the unstable dynamics

$$\dot{x}_u = \eta\, (x_u - x_z).$$

Despite this instability, x_c will remain bounded with respect to x_z provided that

$$x_u^k = \eta \int_{t_k}^{\infty} e^{-\eta(\tau - t_k)} x_z(\tau) d\tau \quad (4)$$

and that $\dot{x}_z$ is bounded. The integral in the right-hand side can be split as the integral from t_k to $t_k + T_c$, which can be readily expressed in terms of the ZMP velocity inputs $\dot{x}_z$ within the MPC control horizon, plus the integral from $t_k + T_c$ to ∞, which depends on the inputs outside the control horizon and therefore makes condition (4) non-causal. To obtain a causal constraint, we compute an approximate value for the second integral by using an *anticipative tail*, i.e., a ZMP trajectory $\tilde{x}_z$ from $t_k + T_c$ to ∞ conjectured on the basis of short-term information from the footstep plan. This leads to the following *stability constraint*

$$\eta \int_{t_k}^{t_{k+C}} e^{-\eta(\tau - t_k)} x_z(\tau) d\tau = x_u^k - \tilde{c}_x^k \quad (5)$$

where

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

Note that the integral in the left-hand side of (5) can be written in closed form as a linear function of the ZMP velocity inputs $\dot{x}_z^k, \ldots, \dot{x}_z^{k+C-1}$.

[2]The variable x_u is also referred to as *capture point* [16] or *divergent component of motion* [1].

Collecting the decision variables over T_c as

$$\dot{X}_z^k = (\dot{x}_z^k, \ldots, \dot{x}_z^{k+C-1}), \quad X_f^k = (x_f^1, \ldots, x_f^F),$$

the k-th iteration of IS-MPC solves the following QP problem, called QP-MPC, for the x component:

$$\begin{cases} \min_{\dot{X}_z^k, X_f^k} \|\dot{X}_z^k\|^2 + \mu \|X_f^k - \hat{X}_f^k\|^2 \\ \text{subject to:} \\ \bullet \text{ ZMP constraints (1), for } j = 0, \ldots, F \\ \bullet \text{ kinematic constraints (3), for } j = 1, \ldots, F \\ \bullet \text{ stability constraint (5)} \end{cases}$$

and an analogous problem for the y component, the only difference being that the kinematic constraint becomes

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

where ℓ is the central value of the distance between right and left feet, and the plus (minus) sign should be used if y_f^j is a footstep for the left (right) foot.

Once a solution is found, the first sample $\dot{x}_z^k, \dot{y}_z^k$ of the optimal input sequence is used to integrate the LIP dynamics along x. The first footstep position x_f^1, y_f^1 is instead used as target landing position for the swing foot in the next single support phase. This results in CoM and swing foot trajectories to be tracked by the humanoid robot with a kinematic controller.

IV. Feasibility-Driven Step Timing Adaptation

In [14] it was shown that IS-MPC is recursively feasible and internally stable (i.e., the CoM trajectory does not diverge with respect to the ZMP) provided that a sufficiently long preview of the footsteps is available. This result was obtained for the case in which the MPC does not modify the footsteps ($\mu = \infty$ in QP-MPC), based on a study of the feasibility region, i.e., the subset of states for which QP-MPC admits a solution.

In this section, we compute the feasibility region for the general case in which the footstep positions are adapted by the MPC, and propose a conservative estimate of such region. Then, we exploit this analysis for step timing adaptation.

A. Feasibility Region

The following proposition characterizes the feasibility region of an IS-MPC iteration in terms of x_u.

Proposition 1: QP-MPC is feasible at t_k if and only if $(x_u^k, y_u^k) \in \mathcal{F}^k$, with

$$\mathcal{F}^k = \{(x_u, y_u) : x_u^{k,m} \leq x_u \leq x_u^{k,M}, y_u^{k,m} \leq y_u \leq y_u^{k,M}\}$$

with the following compact expression for $x_u^{k,m}$ and $x_u^{k,M}$

$$x_u^{k,m/M} = \eta \int_{t_k}^{t_{k+C}} e^{-\eta(\tau - t_k)} x_z^{m/M}(\tau) d\tau + \tilde{c}_x^k \quad (6)$$

$$y_u^{k,m/M} = \eta \int_{t_k}^{t_{k+C}} e^{-\eta(\tau - t_k)} y_z^{m/M}(\tau) d\tau + \tilde{c}_y^k \quad (7)$$

where

$$x_z^m(t) = x_f^{-1} + (x_f^0 - x_f^{-1})\sigma(t, t_s^0, t_s^0 + T_{ds}^0) - \frac{d_{z,x}}{2} - \frac{d_{a,x}}{2}\sum_{j=1}^{F}\sigma(t, t_s^j, t_s^j + T_{ds}^j) \quad (8)$$

$$x_z^M(t) = x_f^{-1} + (x_f^0 - x_f^{-1})\sigma(t, t_s^0, t_s^0 + T_{ds}^0) + \frac{d_{z,x}}{2} + \frac{d_{a,x}}{2}\sum_{j=1}^{F}\sigma(t, t_s^j, t_s^j + T_{ds}^j) \quad (9)$$

$$y_z^m(t) = y_f^{-1} + (y_f^0 - y_f^{-1})\sigma(t, t_s^0, t_s^0 + T_{ds}^0) - \frac{d_{z,y}}{2} - \sum_{j=1}^{F}\left(\frac{d_{a,y}}{2} \mp \ell\right)\sigma(t, t_s^j, t_s^j + T_{ds}^j) \quad (10)$$

$$y_z^M(t) = y_f^{-1} + (y_f^0 - y_f^{-1})\sigma(t, t_s^0, t_s^0 + T_{ds}^0) + \frac{d_{z,y}}{2} + \sum_{j=1}^{F}\left(\frac{d_{a,y}}{2} \pm \ell\right)\sigma(t, t_s^j, t_s^j + T_{ds}^j). \quad (11)$$

Proof. First consider the x coordinate. We start by noticing that all ZMP trajectories produced by IS-MPC satisfy

$$x_z^m \le x_z \le x_z^M, \quad (12)$$

where x_z^m and x_z^M are given by (8) and (9), respectively. This is easily shown by iterating the ZMP constraint (1) to write an inequality on x_z valid throughout the control horizon and then using kinematic constraint (3) to eliminate the footstep positions x_f^j for $j = 1, \dots, F$ from this inequality. In this way, x_z^m and x_z^M represent bounds on x_z accounting for any possible choice of the footstep location (see Fig. 3).

Now multiply each term of (12) by $\eta\, e^{\eta(t-t_k)}$, integrate over time from t_k to t_{k+C}, add $\tilde{c}_x^k$ on each side, and note that the middle term of the resulting inequality is equal to x_u^k thanks to (5), while the outer terms are $x_u^{k,m}$ and $x_u^{k,M}$ as defined in (6). This shows the necessity of the thesis.

To prove sufficiency, assume that $(x_u^k, y_u^k) \in \mathcal{F}^k$, so that x_u^k is a convex combination of the feasibility region bounds:

$$x_u^k = a x_u^{k,m} + (1-a)x_u^{k,M} \quad \text{with } a \in [0,1]\,. \quad (13)$$

Now consider the particular ZMP trajectory

$$x_z^*(t) = a x_z^m(t) + (1-a)x_z^M(t).$$

This trajectory satisfies the ZMP constraint in $[t_k, t_{k+C}]$ for the kinematically admissible sequence of footsteps

$$x_f^j = x_f^{j-1} + a\frac{d_{a,x}}{2}.$$

It also satisfies the stability constraint (5), since

$$x_u^k = \eta\int_{t_k}^{t_{k+C}} e^{-\eta(\tau - t_k)} x_z^*(\tau)d\tau + \tilde{c}_x^k = a x_u^{k,m} + (1-a)x_u^{k,M}$$

is true thanks to (13). We have thus shown that there exists at least one ZMP trajectory and a sequence of footsteps satisfying all constraints.

The proof for y is identical except for the footstep position, which must incorporate the lateral displacement

$$y_f^j = y_f^{j-1} + a\frac{d_{a,y}}{2} \pm \ell.$$

■

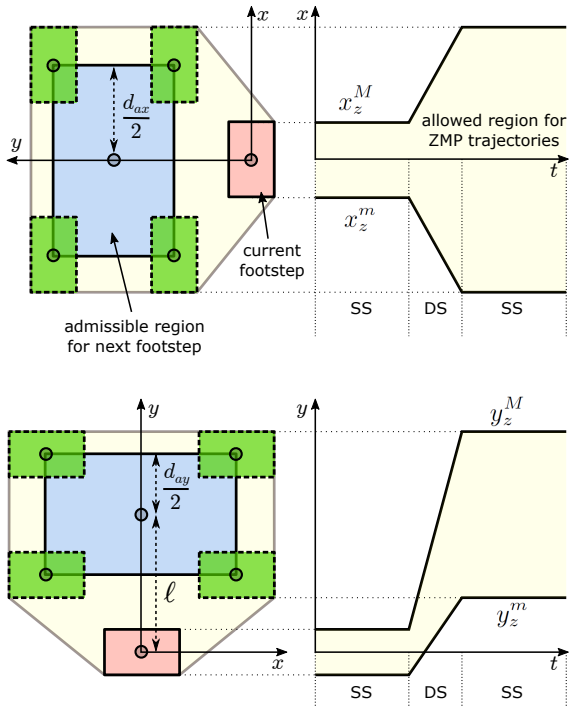

Fig. 3. A visual representation of the kinematic constraint (left) and how it translates to the bounds on the ZMP trajectory expressed by (8–11) (right). The green footsteps stand at the extreme positions compatible with the kinematic constraint (blue region).

The bounds (6–7) of the feasibility region $\mathcal{F}^k$ can be computed in closed form. To this end, we replace step durations with more convenient variables (see Fig. 2): the remaining duration of the *current* step (called in the following *time-to-step*) will be represented by

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

whereas the duration of the *subsequent* steps will be encoded in $\lambda_{1\text{-}F} = \{\lambda_1, \dots, \lambda_F\}$, where

$$\lambda_j = e^{-\eta T_s^j}.$$

Also, let $\nu = T_{ds}^j / T_s^j$ be the ratio of the duration of double support over that of the whole step, which in the prediction is assumed constant for subsequent steps ($j = 1, \dots, F$).

In the above variables, bounds (6–7) assume the following expression for t_k in a double support phase

$$x_u^{k,m/M} = \alpha_{ds,x}^{m/M}(\lambda_{1\text{-}F})\Delta\lambda + \beta_{ds,x}^{m/M}\log\Delta\lambda + \gamma_{ds,x}^{m/M} \quad (14)$$

$$y_u^{k,m/M} = \alpha_{ds,y}^{m/M}(\lambda_{1\text{-}F})\Delta\lambda + \beta_{ds,y}^{m/M}\log\Delta\lambda + \gamma_{ds,y}^{m/M} \quad (15)$$

while for t_k in a single support phase we obtain

$$x_u^{k,m/M} = \alpha_{ss,x}^{m/M}(\lambda_{1\text{-}F})\Delta\lambda + \gamma_{ss,x}^{m/M} \quad (16)$$

$$y_u^{k,m/M} = \alpha_{ss,y}^{m/M}(\lambda_{1\text{-}F})\Delta\lambda + \gamma_{ss,y}^{m/M}. \quad (17)$$

The expressions of the α, β, γ coefficients in the above formulas, as well as for the explicit computations leading to

them, are omitted for lack of space. As an example, we provide here the coefficients appearing in (14) for $x_u^{k,m}$:

$$\begin{aligned}
\alpha_{ds,x}^m(\lambda_{1\text{-}F}) &= \frac{x_f^{-1}-x_f^0}{\eta T_{ds}^0}e^{\eta T_{ss}^0}+\frac{d_{a,x}}{2}\sum_{j=1}^{F}\frac{1-\lambda_j^\nu}{\nu\log\lambda_j}\prod_{i=1}^{j-1}\lambda_i \\
\beta_{ds,x}^m &= \frac{x_f^0-x_f^{-1}}{\eta T_{ds}^0} \\
\gamma_{ds,x}^m &= (x_f^0-x_f^{-1})\frac{1+\eta T_s^0}{\eta T_{ds}^0}-\frac{d_{z,x}}{2}(1-e^{-\eta T_c}) \\
&\quad +x_f^{-1}-x_f^0 e^{-\eta T_c}+F\frac{d_{a,x}}{2}e^{-\eta T_c}+\tilde{c}_x^k.
\end{aligned}$$

B. Conservative Estimate of the Feasibility Region

Formulas (14–17) indicate that the bounds of the feasibility region depend on the durations $\lambda_1,\dots\lambda_F$ of the subsequent steps; this dependence, as shown by the above expressions of the α coefficients, is actually nonlinear. Therefore, to make the step timing adaptation problem viable, $\lambda_1,\dots\lambda_F$ will be left unchanged, while the time-to-step $\Delta\lambda$ can take any value within a suitable interval $[\Delta\lambda^{\min},\Delta\lambda^{\max}]$. This simplification is reasonable since the current step can be expected to be the most critical for absorbing a perturbation.

To remove the nonlinearity in $\Delta\lambda$ from the double support bounds (14–15), consider that $\log\Delta\lambda$ in $[\Delta\lambda^{\min},\Delta\lambda^{\max}]$ can be bounded above by the tangent computed at $\Delta\lambda^{\text{mid}}=(\Delta\lambda^{\min}+\Delta\lambda^{\max})/2$, and below by the chord joining $\log\Delta\lambda^{\min}$ and $\log\Delta\lambda^{\max}$. By doing so, one obtains the following inequalities for the bounds of $\mathcal{F}^k$ in double support:

$$\begin{aligned}
x_u^{k,m} &\le \bar{\alpha}_{ds,x}^m\Delta\lambda+\bar{\gamma}_{ds,x}^m=\bar{x}_u^{k,m} \\
x_u^{k,M} &\ge \bar{\alpha}_{ds,x}^M\Delta\lambda+\bar{\gamma}_{ds,x}^M=\bar{x}_u^{k,M} \\
y_u^{k,m} &\le \bar{\alpha}_{ds,y}^m\Delta\lambda+\bar{\gamma}_{ds,y}^m=\bar{y}_u^{k,m} \\
y_u^{k,M} &\ge \bar{\alpha}_{ds,y}^M\Delta\lambda+\bar{\gamma}_{ds,y}^M=y_u^{k,M},
\end{aligned}$$

where

$$\begin{aligned}
\bar{\alpha}_{ds,x}^m &= \alpha_{ds,x}^m+\frac{\beta_{ds,x}^m}{\Delta\lambda^{\text{mid}}} \\
\bar{\gamma}_{ds,x}^m &= \gamma_{ds,x}^m+\beta_{ds,x}^m\left(\log\Delta\lambda^{\text{mid}}-1\right).
\end{aligned}$$

and similarly for the other coefficients. Based on this, we define a conservative estimate of $\mathcal{F}^k$ as

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

Note that $\mathcal{F}_{\text{est}}^k=\mathcal{F}^k$ during the single support phase.

Figure 4 shows the evolution over time of the bounds of $\mathcal{F}^k$ vs. those of $\mathcal{F}_{\text{est}}^k$ over two steps of duration 0.5 s each, with $T_{ds}=0.2$ s and $T_{ds}=0.3$ s, for the LIP model described in Sect. V-A. As expected, $\mathcal{F}_{\text{est}}^k$ accurately approximates $\mathcal{F}^k$, with which it coincides in single support.

Since $\mathcal{F}^k$ is a rectangular region of dimensions $x_u^{k,M}-x_u^{k,m}$ and $y_u^{k,M}-y_u^{k,m}$, Fig. 4 also implies that its extension monotonically increases during single support, consistently with the fact that the allowed ZMP region (the yellow area in Fig. 3) grows over time, and abruptly decreases at the start of double support, due to the reset of the ZMP trajectory bounds to the boundaries of the (new) support foot. See also the animation of the feasibility region in Simulation 1 of the accompanying video.

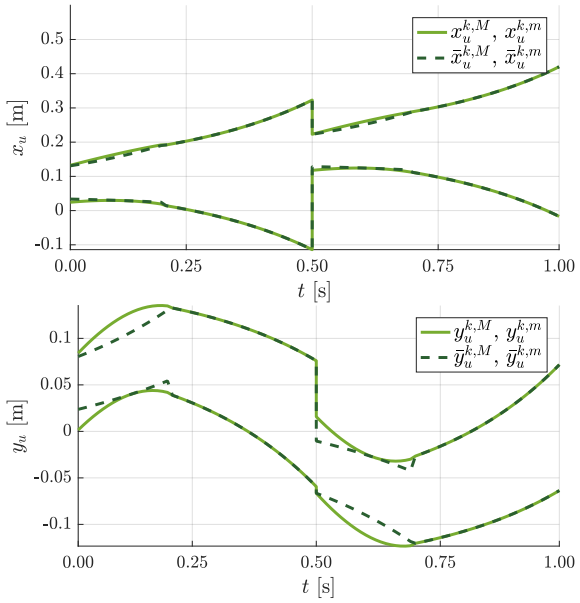

Fig. 4. Time evolution of the bounds of $\mathcal{F}^k$ (solid) vs. $\mathcal{F}_{\text{est}}^k$ (dashed).

C. Step Timing Adaptation (STA)

In principle, the STA stage should intervene whenever (x_u^k,y_u^k) moves outside of the feasibility region $\mathcal{F}^k$ due to a perturbation, such as a push on the robot. In practice, leveraging the results of Sect. IV-B, we replace $\mathcal{F}^k$ with $\mathcal{F}_{\text{est}}^k$ to take advantage of the linearity of its bounds in $\Delta\lambda$.

At each iteration the following QP problem, called QP-STA, is solved to compute the adapted duration of the current step *before* sending the timing vector to IS-MPC (see Fig. 1):

$$\min_{\Delta\lambda}(\Delta\lambda-\widehat{\Delta\lambda})^2$$

subject to:

- feasibility constraints for x_u and y_u

$$\begin{aligned}
\bar{x}_u^{k,m}+\epsilon_x &\le x_u^k\le\bar{x}_u^{k,M}-\epsilon_x \\
\bar{y}_u^{k,m}+\epsilon_y &\le y_u^k\le\bar{y}_u^{k,M}-\epsilon_y
\end{aligned}$$

- timing constraints

$$\begin{aligned}
e^{-\eta(t_s^0+T_s^{\max}-t_k)}\le\Delta\lambda&\le e^{-\eta(t_s^0+T_s^{\min}-t_k)} \\
\Delta\lambda&\le e^{-\eta\epsilon_t}
\end{aligned}$$

in which:

- $\widehat{\Delta\lambda}=e^{-\eta(\hat{t}_s^1-t_k)}$ is the time-to-step (in exponential encoding) according to the current candidate timing;
- ϵ_x and ϵ_y are (positive) safety margins for the feasibility constraints, introduced to push solutions *inside* $\mathcal{F}_{\text{est}}^k$;
- $\epsilon_t=\begin{cases}\hat{T}_{ss}^0+\max\left(|\zeta_x^k-x_f^0|,|\zeta_y^k-y_f^0|\right)/v_z^{\max} & \text{(DS)}\\ \max\left(|x_{\text{sw}}^k-x_f^{1|k-1}|,|y_{\text{sw}}^k-y_f^{1|k-1}|\right)/v_{\text{sw}}^{\max} & \text{(SS)}\end{cases}$

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

The first timing constraint of QP-STA means that any step duration should be within $T_s^{\min}$ and $T_s^{\max}$, limit values that typically depend on the robot. The second constraint reflects the fact that only the duration of the current phase (SS or DS) is adapted, so that in double support the time-to-step cannot be reduced below the candidate duration of single support; it also guarantees that the robot has enough time to complete the current phase (similarly to [4]) given the bounds $v_z^{\max}$ on the ZMP velocity (DS) and $v_{\text{sw}}^{\max}$ on the swinging foot velocity (SS), assumed to be the same over x and y.

Since QP-STA is formulated as a least squares problem, the current step timing will indeed be modified only if (x_u^k, y_u^k) is outside $\mathcal{F}_{\text{est}}^k$. In this case, denoting by $\Delta\lambda \neq \widehat{\Delta\lambda}$ the solution of QP-STA, the adapted timestamp of the next step is computed as $t_s^1 = t_k + \log(1/\Delta\lambda)/\eta$.

To get an intuitive understanding of how step timing adaptation works, look at Fig. 5, which shows the effects on $\mathcal{F}_{\text{est}}^k$ of both a reduction and an extension of the current step duration at a certain time $\bar{t}$. For example, assume that a push in the x direction (positive or negative) occurs at $\bar{t}$, so that x_u will experience a displacement in the same direction; in this case, QP-STA will reduce the current step duration so as to enlarge $\mathcal{F}_{\text{est}}^k$ along x. The same kind of adaptation will work for a lateral push directed from the support towards the swing leg (positive y), because $\mathcal{F}_{\text{est}}^k$ will be shifted up in y.

The situation changes for a lateral push from the swing towards the support leg (negative y). In this case, Fig. 5 indicates that step duration should be extended, as this will shift $\mathcal{F}_{\text{est}}^k$ down in y. We will come back on this in Sect. V-C.

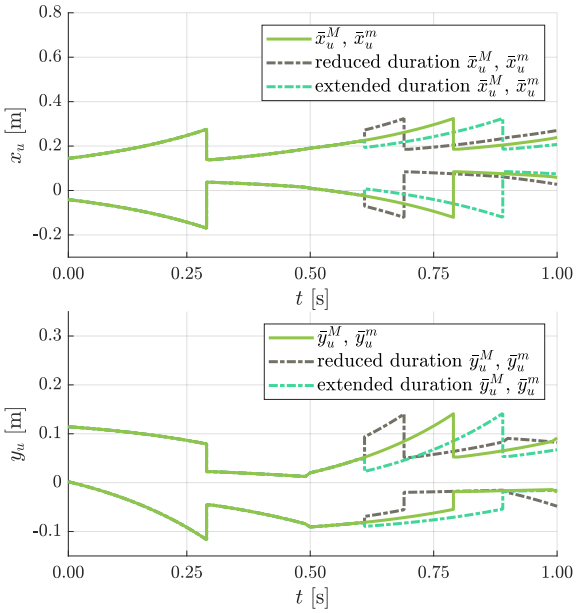

Fig. 5. Effect of step timing adaptation on feasibility: modification of the bounds of $\mathcal{F}_{\text{est}}^k$. Here, the robot is walking in the positive x direction when at time $\bar{t} = 0.6$ s, with the left foot swinging, the duration of the current step is reduced/ extended by 0.1 s.

V. Simulations on the LIP

We now showcase the performance of our STA method by means of some simulations for the LIP.

A. Push Recovery

The benefit of STA in conjunction with IS-MPC can be appreciated in a push recovery scenario on the LIP. We use the following parameters: $m = 39$ kg, $\bar{z}_c = 0.75$ m, $d_{z,x} = d_{z,y} = 0.08$ m, $d_{a,x} = 0.5$ m, $d_{a,y} = 0.16$ m, $\ell = 0.2$ m for the robot (corresponding to the HRP-4 humanoid), and $T_c = 1$ s, $\delta = 0.01$ s, $T_s^{\min} = 0.2$ s, $T_s^{\max} = 0.65$ s, $\epsilon_x = \epsilon_y = 0.005$ m, $v_z^{\max} = 1.5$ m/s, $v_{\text{sw}}^{\max} = 1$ m/s for the QP problems. In the candidate timing, all steps have duration $\hat{T}_s = 0.5$ s with DS phase lasting $\hat{T}_{ds} = 0.2$ s. The simulation is run in MATLAB using `quadprog` as QP solver.

The robot is walking under the action of IS-MPC when at $t = 2.7$ s (i.e., at the start of single support of the sixth step) it receives a diagonal push $F_{\text{ext}} = (97.5, -136.5, 0)$ [N] lasting 0.1 s. Figure 6 shows the situation at the end of the push: without STA, the current value of (x_u, y_u) would be outside the feasibility region, leading to internal instability and failure, whereas STA modifies the feasibility region of IS-MPC in such a way to contain (x_u, y_u). In particular, this is obtained by reducing the duration of the current step by 0.13 s. To fully absorb the effect of the push, it is also necessary to reduce the next double support by 0.14 s. In fact, although step timing adaptation allows to maintain feasibility until the end of the push, adaptation at successive instants is still required because QP-STA sees in this case a violation of the conservative estimate $\mathcal{F}_{\text{est}}$. See the accompanying video for a complete clip of this simulation.

We repeated this simulation for different pushing forces, without changing the instant of application. IS-MPC with STA withstands x-forces in $[-260, 304.2]$ N and y-forces in $[-273, 124.8]$ N. Without STA, these ranges are reduced to $[-136.5, 144.2]$ N and $[-93.6, 97.5]$ N, respectively.

B. Comparison with DCM-Based Method

The next simulation compares the proposed approach with DCM-based step timing adaptation, identified as the state of the art in the field. In particular, we selected the technique presented in [10], which uses ∞-step capturability bounds to formulate a QP with footstep positions and timing as decision variables; futhermore, the ZMP is assumed to be always at the center of the support foot (i.e., the robot has point feet), and double support is instantaneous. The parameters used for the comparison are the same as the previous simulation, with the exception of the $d_{z,x}$ and $d_{z,y}$ which were reduced to 0.01 m for IS-MPC, to mimic a point-foot situation.

The robot is walking when, at the start of a single support, it receives a lateral push $F_{\text{ext}} = (0, 226.2, 0)^T$ [N] for 0.1 s. The results of the comparison are shown in Fig. 7. While IS-MPC with STA absorbs the push and quickly resume forward walking, the DCM-based method fails due to a loss of feasibility, which leads to divergence between the CoM and the ZMP (see the video). With the reduced footprint of this simulation, IS-MPC with STA can withstand x-forces in $[-288.6, 290]$ N and y-forces in $[-46.8, 245.7]$ N. Again, this is better than the DCM-based method, for which the ranges are $[-265.2, 265.2]$ N and $[-31.2, 222.3]$ N.

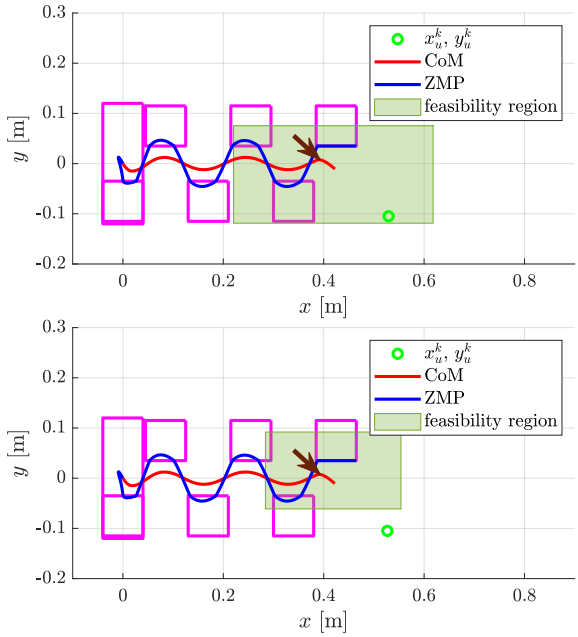

Fig. 6. LIP simulation 1: Push recovery while walking. Feasibility region of IS-MPC at the end of the push ($t = 2.8$ s) with STA (top) and without STA (bottom). Footsteps are shown in magenta. Note how STA allows to enlarge the feasibility region so as to keep the current (x_u, y_u) inside it.

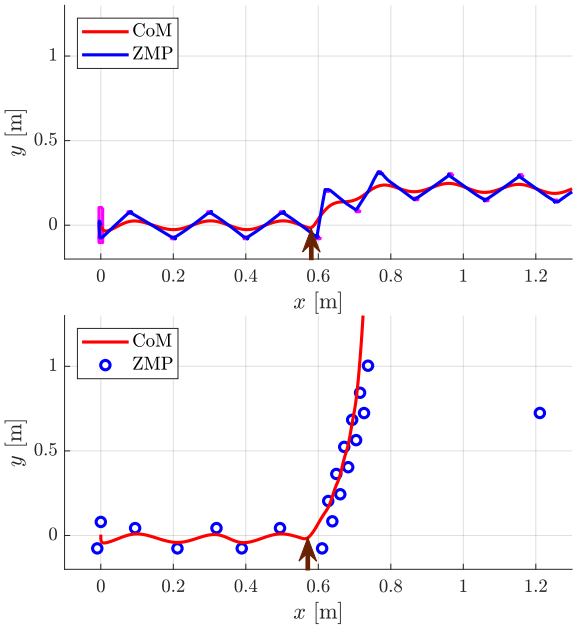

Fig. 7. LIP simulation 2: Push recovery while walking. IS-MPC with STA (top) vs. DCM-based method [10] (bottom). IS-MPC with STA withstands the push while the DCM-based technique exhibits an unstable behavior.

The reason for the superior performance of IS-MPC with STA over the DCM-based method is twofold. First, it appears that our method can take full advantage of finite-size feet, albeit small, and of the non-instantaneous double support. Even more importantly, while our stability constraint is the correct condition for avoiding CoM/ZMP divergence *at any time*, the ∞-step capturability constraint of [10] is only appropriate *at the start of each step*[3], so that its use at intermediate instants may lead to failure in critical cases.

C. Discussion

The results of this section show that modifying step timings is an effective way to counteract an imminent loss of feasibility. With respect to other techniques for time adaptation, our approach is designed to work in conjunction with IS-MPC, and thus inherits the favorable properties guaranteed by the use of preview information in the stability constraint [14].

The analytical expressions (14–17) of the feasibility region bounds can in principle be used to set up a nonlinear optimization problem to adapt multiple steps in the control horizon, as opposed to the current step only. Due to the exponentially diminishing effect of future steps in eqs. (6-7), however, the benefit of such strategy is bound to be limited.

There is, however, at least one case in which adapting a future step is a sensible option. As discussed at the end of Sect. IV-C, if a lateral push is directed from the swing towards the support leg, QP-STA reacts by increasing the duration of the current step (see Fig. 5). This is somewhat counterintuitive, as the appropriate reaction would be to displace the footstep in the same direction to keep the (perturbed) ZMP within the support polygon. However, IS-MPC does not contemplate such a possibility, because the kinematic constraint prevents placing one foot in front of the other, and this is reflected in $\mathcal{F}^k$; therefore, QP-STA will extend as much as possible the duration of the current single support, as putting the foot down will necessarily displace the support polygon in the wrong direction. However, in the case of very strong swing-to-support pushes, QP-STA may be unable to maintain feasibility by extending the duration of the current step. In this case, a better strategy would be to put down the swinging foot as soon as possible and recover by adapting the next step. This can be enforced as a rule-based behavior whenever QP-STA is unfeasible. See the accompanying video for a simulation.

[3]It is possible to prove (details are omitted) that the feasibility region $\mathcal{F}^k$ of IS-MPC at the start of a step tends to coincide with the ∞-step capturability region of [10] if, in addition to point feet and instantaneous double support, we also assume that the control horizon approaches infinity.

VI. Dynamic Simulations

The proposed approach has been validated for the HRP-4 humanoid through dynamic simulations in the DART environment, using the same parameters of the LIP simulations. Both QP-MPC and QP-STA are solved using HPIPM [17]. In particular, setting up and solving QP-STA takes about 60 μs, which is more than appropriate for real-time implementation.

Three different scenarios are considered: forward walking, backward diagonal walking and walking on the spot. In each simulation, IS-MPC with STA is shown to withstand pushes that would cause the robot to fall without STA. Figure 8 shows snapshots of the forward walking simulation. Clips of all dynamic simulations are included in the accompanying video.

VII. Experiments

Experiments on a NAO were performed to validate the proposed approach on a physical platform. We used the following

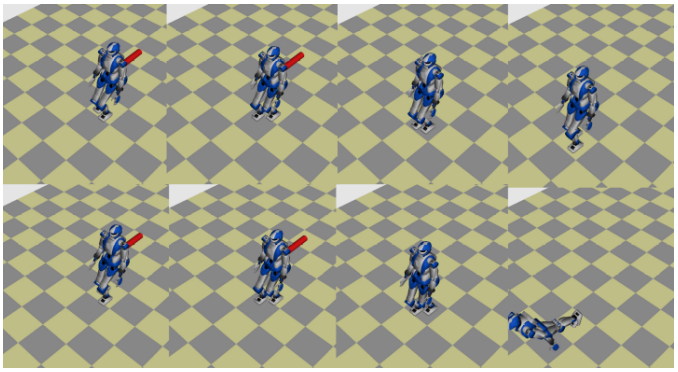

Fig. 8. HRP-4 dynamic simulation 1: Push recovery while walking forward. IS-MPC with STA (top row) and without STA (bottom row). In the presence of a push from the back, HRP-4 falls if no timing adaptation is performed.

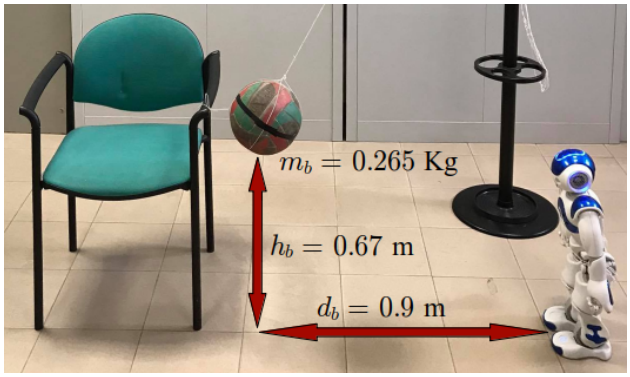

Fig. 9. The pendulum used in the experiments to apply a replicable push. See the accompanying video for clips of the experiments.

parameters: $m = 5.3$ kg, $\bar{z}_c = 0.245$ 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, $\delta = 0.05$ s, $T_s^{\min} = 0.3$ s All other parameters are the same of the previous simulation. The pendulum structure of Fig. 9 was set up for applying a replicable push to the robot.

The accompanying video shows NAO receiving a frontal push when walking. Consistently with the simulations, use of IS-MPC with STA allows the robot to withstand the perturbation, whereas a fall occurs when step timings are not adapted. Over a campaign of 10 experiments, IS-MPC with STA had a 90% success rate, whereas IS-MPC without STA recovered in only 20% of cases.

VIII. Conclusions

Step time adaptation is a way for achieving robust locomotion in humanoids. Based on a feasibility analysis of our IS-MPC method for stable gait generation, we have designed a step timing adapter that modifies the duration of the current step whenever a loss of feasibility is imminent due to a perturbation. The proposed approach, which allows the IS-MPC algorithm to maintain its linearity, has been validated by both simulations and experiments. We are now working on a robust framework that uses STA in conjunction with disturbance observation [18] and ZMP constraint restriction [19].

At the same time this paper was submitted, a preprint appeared where feasibility is used for robust gait generation [20]; in particular, unfeasible states are artificially projected inside the feasibility region before using them in the MPC. This is conceptually very different from our approach, where the step timing is adapted so as to *maintain* feasibility of the current state, which is never modified.

References

[1] T. Takenaka, T. Matsumoto, and T. Yoshiike, "Real time motion generation and control for biped robot - 1st report: Walking gait pattern generation," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2009, pp. 1084–1091.

[2] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, "Biped walking pattern generation by using preview control of zero-moment point," in *IEEE Int. Conf. on Robotics and Automation*, 2003, pp. 1620–1626.

[3] 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.

[4] A. Herdt, H. Diedam, P.-B. Wieber, D. Dimitrov, K. Mombaur, and M. Diehl, "Online walking motion generation with automatic footstep placement," *Advanced Robotics*, vol. 24, no. 5-6, pp. 719–737, 2010.

[5] N. Bohórquez and 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.

[6] S. Caron and 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.

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

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

[9] M. R. O. A. Maximo, C. H. C. Ribeiro, and R. J. M. Afonso, "Mixed-integer programming for automatic walking step duration," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2016, pp. 5399–5404.

[10] M. Khadiv, A. Herzog, S. A. A. Moosavian, and L. Righetti, "Step timing adjustment: A step toward generating robust gaits," in *16th IEEE-RAS Int. Conf. on Humanoid Robots*, 2016, pp. 35–42.

[11] M. Shafiee, G. Romualdi, S. Dafarra, F. J. A. Chavez, and 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.

[12] H. Jeong, I. Lee, J. Oh, K. K. Lee, and J.-H. Oh, "A robust walking controller based on online optimization of ankle, hip, and stepping strategies," *IEEE Transactions on Robotics*, vol. 35, no. 6, pp. 1367–1386, 2019.

[13] R. J. Griffin, G. Wiedebach, S. Bertrand, A. Leonessa, and J. Pratt, "Walking stabilization using step timing and location adjustment on the humanoid robot, Atlas," in *IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2017, pp. 667–673.

[14] N. Scianca, D. De Simone, L. Lanari, and G. Oriolo, "MPC for humanoid gait generation: Stability and feasibility," *IEEE Transactions on Robotics*, vol. 36, no. 4, pp. 1171–1188, 2020.

[15] P. Ferrari, N. Scianca, L. Lanari, and G. Oriolo, "An integrated motion planner/controller for humanoid robots on uneven ground," in *18th European Control Conf.*, 2019, pp. 1598–1603.

[16] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture point: A step toward humanoid push recovery," in *6th IEEE-RAS Int. Conf. on Humanoid Robots*, 2006, pp. 200–207.

[17] G. Frison, in *https://github.com/giaf/hpipm*, 2020.

[18] F. M. Smaldone, N. Scianca, V. Modugno, L. Lanari, and 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.

[19] ——, "ZMP constraint restriction for robust gait generation in humanoids," in *2020 IEEE Int. Conf. on Robotics and Automation*, 2020, pp. 651–657.

[20] M. H. Yeganegi, M. Khadiv, A. Del Prete, S. A. A. Moosavian, and L. Righetti, "Robust walking based on MPC with viability-based feasibility guarantees," in https://arxiv.org/abs/2010.04514, 2020.