2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

Joint-Level IS-MPC: a Whole-Body MPC with Centroidal Feasibility for Humanoid Locomotion

Tommaso Belvedere, Nicola Scianca, Leonardo Lanari, and Giuseppe Oriolo

Abstract— **We propose an effective whole-body MPC controller for locomotion of humanoid robots. Our method generates motions using the full kinematics, allowing it to account for joint limits and to exploit upper-body motions to reject disturbances. Each MPC iteration solves a single QP that considers the interplay between dynamic and kinematic features of the robot. Thanks to our special formulation, we are able to perform a feasibility analysis, which opens the door to future enhancements of functionality and performance, e.g., step adaptation in complex environments. We demonstrate its effectiveness through a campaign of dynamic simulations aimed at highlighting how the joint limits and the use of the angular momentum through upper-body motions are fundamental for maximizing performance, robustness, and ultimately make the robot able to execute more challenging gaits.**

I. Introduction

Interest in humanoid robots has been on the rise, with companies now working towards producing more affordable models for the consumer market. This growing fascination is fueled by advancements in actuator technology, offering increased power density and reduced gear ratios. Additionally, the popularity surge is also driven by advances on the control side. Whereas reinforcement learning techniques have been often successfully applied on quadrupeds [1], Model Predictive Control (MPC) remains the go-to option for humanoids. In MPC, control actions are determined online by solving a constrained optimization problem over a prediction horizon. However, the complexity of the problem requires some compromises to make the application to humanoid robots tractable.

Historically, the most popular approaches used simplified models to reduce the number of optimization variables. Among these, the most common is the Linear Inverted Pendulum (LIP) model [2], which relates the dynamics of the Center of Mass (CoM) to the Zero Moment Point (ZMP). The ZMP is the point of application of the resultant ground reaction force, and must be kept within the support polygon (the convex hull of contact surfaces) in order to maintain balance. While such approaches have proven to be effective, they have two main drawbacks: *(i)* they neglect dynamic effects such as angular momentum variations, which might be relevant when performing faster motions, and whose omission can limit the performance; *(ii)* they require a separate whole-body controller to realize the high-level trajectories planned by the MPC [3], [4], [5], whose commands could be inconsistent with the MPC constraints.

A radically different approach is to perform whole-body MPC using the full model of the robot [6]. This is a complex endeavor because humanoids have a large number of degrees of freedom, their dynamics are highly nonlinear, and the ability to engage and disengage contacts makes this effectively a hybrid dynamics problem. To this day, there exist only few works that showcase real-time whole-body MPC for humanoid robots, even in simulation (see [7] for a survey of recent results). A promising approach uses Differential Dynamic Programming (DDP) [8] which enables faster computations when using the full model, but forgoes the ability to enforce hard constraints. Moreover, since many humanoid robots are controlled in position with high-gain servos [9], in many cases it might be unnecessary to solve the whole-body dynamics as the resulting torques cannot be directly used for control.

To reduce the number of optimization variables it is possible to use models of different complexity at different stages across the MPC prediction horizon [10]. In [11], this technique was successfully applied to whole-body humanoid control, highlighting how upper-body motions can be essential to reject disturbances.

Another interesting way to scale down the complexity of the problem is to consider a model composed of the centroidal dynamics and the full kinematics [12], thus reducing the nonlinearities. In this work we use a similar approach.

In previous works, we have proposed Intrinsically Stable MPC (IS-MPC) [13], which is a LIP-based MPC scheme that enforces an explicit *stability constraint* on the unstable part of the dynamics. Thanks to the addition of this constraint we were able to study the feasibility properties of the scheme, which offers guarantees, but more importantly allows to devise several functional improvements, such as feasibility-based step timing adaptation [14], and more general online footstep adaptation techniques [15], [16].

In this work, we propose *Joint-level IS-MPC* (JIS-MPC): a whole-body MPC scheme taking joint accelerations as control inputs on the robot. We define a set of centroidal dynamic and kinematic tasks that allows the robot to produce a stable gait, and show an efficient way of formulating the associated Quadratic Program (QP). We enforce joint limits,

Tommaso Belvedere is with CNRS, Univ Rennes, Inria, IRISA, Campus de Beaulieu, 35042 Rennes Cedex, France. E-mail: tommaso.belvedere@irisa.fr

Nicola Scianca, Leonardo Lanari and Giuseppe Oriolo 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.

This work was carried out while Tommaso Belvedere was a a Ph.D. student at the Dipartimento di Ingegneria Informatica, Automatica e Gestionale, Sapienza Università di Roma.

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

979-8-3503-7769-9/24/$31.00 ©2024 IEEE

ZMP and stability constraints, and perform a feasibility study that opens the possibility to use the feasibility-based tools we developed for standard IS-MPC on the proposed method with minimal modifications.

We believe our approach offers a series of advantages with respect to the state of the art. For example, when comparing to [11], we do not need to resort to a simplified model to increase the length of the horizon, and we include joint position constraints that are shown to be essential when preforming challenging gaits. Compared to [8], we are able to generate whole-body motions including the upper body (torso and arms), instead of just the lower body, and show how this is useful for disturbance rejection. Moreover, we are not subject to the limitations of DDP in enforcing hard constraints.

We validate the effectiveness of the proposed method with dynamic simulations on the Kawada HRP-4 robot. We show that, compared to a standard IS-MPC using the simplified model, JIS-MPC is able to produce substantially faster gaits, both in terms of stride length and of step timing. It is also able to better tolerate disturbances, thanks to its ability to use angular momentum to more accurately determine the position of the ZMP. In order to prove the usefulness of studying the feasibility properties, we show that tracking the center of the feasibility region produces better results rather than tracking the center of the ZMP constraint, which is the standard alternative.

The paper is organized as follows. Section II describes in general terms the proposed JIS-MPC control scheme, while Section III details the prediction model. The constraints presented in Section IV are enforced in the MPC, see Section V. These are used in the feasibility analysis provided in Section VI while simulation results are discussed in Section VII. Finally, some future work is mentioned in Section VIII.

II. The proposed approach

Consider the problem of generating joint commands for the humanoid robot so as to execute an assigned footstep plan, i.e., step over a sequence of given footsteps according to a specified timing. To this end, we propose an MPC scheme with a mixed kinematic-dynamic prediction model. On the one hand, we use *centroidal dynamics* [17] to relate the variation of the linear and angular momentum of the humanoid to the position of the ZMP, which is featured in the balance condition. On the other hand, the kinematic part of the model expresses all quantities of interest in terms of joint variables, which are taken as decision variables to account for the mechanical limitations of the actual robot, such as joint limits.

To successfully achieve locomotion, we define a set of tasks that will be either enforced as constraints or tackled via the cost function of our MPC formulation:

- keeping the ZMP inside the support polygon;
- keeping the support foot still on the ground;
- executing the swing foot trajectory;
- ensuring that the resulting CoM trajectory is bounded with respect to the ZMP.

Additional tasks are considered, which may either improve the quality of the resulting gait, such as

- keeping an upright torso orientation;
- controlling the vertical angular momentum to reduce the vertical component of contact torques, limiting the chance of slipping;

or allow the robot to perform other operations, e.g.,

- manipulating an object;
- controlling the head of the robot to orient its gaze.

The only control inputs of our model are joint accelerations. While in the literature contact forces are commonly included among the decision variables, we enforce dynamic consistency via a constraint on the ZMP. This has the advantage of reducing the number of variables; in fact, contact wrenches are 6-dimensional vectors which, once propagated over the entire horizon, can amount to hundreds of additional decision variables. Clearly, with this choice we forgo the possibility of enforcing friction constraints; however, in our experience, friction is rarely a limiting factor when walking on flat ground, so that we reserve this extension for future works in more challenging environments.

Joint accelerations are mapped to the different tasks via linearized differential kinematics. This allows to formulate our MPC optimization problem as a QP, with obvious advantages in computational efficiency. The resulting QP is in condensed form, i.e., states do not appear as decision variables.

In the proposed approach, it is assumed that the footstep plan is generated by an external module and not altered by the MPC. To preserve the reactivity of the gait generation framework in the presence of perturbations, we could rely on the use of a real-time planner [16] which adapts footsteps and timing so as to always guarantee feasibility of the MPC; more on this in the concluding section.

III. Modeling

In this section, we introduce prediction models for both the robot and the various tasks of interest.

A. Dynamic model

Consider a humanoid robot with n revolute joints. Its configuration is described by $\boldsymbol{q} = (\boldsymbol{q}_b, \boldsymbol{q}_j)$, where $\boldsymbol{q}_b \in SE(3)$ is the pose of the floating base (e.g., the torso) and $\boldsymbol{q}_j \in (SO(2))^n$ is the configuration of the joints. Moreover, let $\boldsymbol{\nu} = (\boldsymbol{v}_b, \boldsymbol{\omega}_b, \dot{\boldsymbol{q}}_j)$ be the (n+6)-dimensional velocity vector, where $\boldsymbol{v}_b$ and $\boldsymbol{\omega}_b$ denote the linear and angular velocities of the floating base.

Since joint accelerations $\ddot{\boldsymbol{q}}_j$ are the control inputs $\boldsymbol{u} \in \mathbb{R}^n$, we can directly write

$$\dot{\boldsymbol{\nu}} = \begin{pmatrix} \boldsymbol{\alpha}_b \\ \boldsymbol{u} \end{pmatrix}, \tag{1}$$

where $\boldsymbol{\alpha}_b$ is the 6-dimensional vector of linear and angular accelerations of the floating base. Clearly, $\boldsymbol{\alpha}_b$ is not directly controllable; its evolution must be consistent with the contact sequence prescribed by the footstep plan. In particular, the pose of the support foot $\boldsymbol{p}_{sf}(\boldsymbol{q})$ is subject at all times to a constraint of the form

$$\boldsymbol{p}_{sf}(\boldsymbol{q}) = \text{constant}, \tag{2}$$

which, together with appropriate conditions on the ZMP, ensures a stable contact [18]. Continuous satisfaction of this constraint leads to

$$\boldsymbol{J}_{sf}(\boldsymbol{q})\dot{\boldsymbol{\nu}} + \dot{\boldsymbol{J}}_{sf}(\boldsymbol{q},\boldsymbol{\nu})\boldsymbol{\nu} = \boldsymbol{0}, \tag{3}$$

where $\boldsymbol{J}_{sf}(\boldsymbol{q})$ is the contact Jacobian, i.e., the $6 \times (n+6)$ matrix such that $\dot{\boldsymbol{p}}_{sf}(\boldsymbol{q}) = \boldsymbol{J}_{sf}(\boldsymbol{q})\boldsymbol{\nu}$. Equations (1) and (3) together express the second-order kinematic model of the robot in contact.

The second part of the model is the centroidal dynamics, i.e., the dynamics of the centroidal momentum $\boldsymbol{h}_c = (m\dot{\boldsymbol{p}}_c, \boldsymbol{l})$, with m the mass of the robot, $\boldsymbol{p}_c = (x_c, y_c, z_c)$ the CoM position, and $\boldsymbol{l} = (l_x, l_y, l_z)$ the angular momentum of the robot at the CoM. We use the centroidal momentum matrix [17] to relate $\boldsymbol{h}_c$ to the joint velocities:

$$\begin{pmatrix} m\dot{\boldsymbol{p}}_c \\ \boldsymbol{l} \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}_c(\boldsymbol{q}) \\ \boldsymbol{A}_l(\boldsymbol{q}) \end{pmatrix} \boldsymbol{\nu}. \tag{4}$$

By differentiating (4), we obtain an expression depending on our control inputs:

$$\begin{pmatrix} m\ddot{\boldsymbol{p}}_c \\ \dot{\boldsymbol{l}} \end{pmatrix} = \begin{pmatrix} \boldsymbol{A}_c(\boldsymbol{q}) \\ \boldsymbol{A}_l(\boldsymbol{q}) \end{pmatrix} \dot{\boldsymbol{\nu}} + \begin{pmatrix} \dot{\boldsymbol{A}}_c(\boldsymbol{q}) \\ \dot{\boldsymbol{A}}_l(\boldsymbol{q}) \end{pmatrix} \boldsymbol{\nu}. \tag{5}$$

Predicting the evolution of the centroidal momentum is necessary for ensuring the dynamic realizability of contacts. In fact, on flat ground, unilaterality of the contact forces requires that the ZMP $\boldsymbol{p}_z = (x_z, y_z, 0)$ is always inside the support polygon. As shown in [2], the position of the ZMP is given by

$$\begin{pmatrix} x_z \\ y_z \end{pmatrix} = \begin{pmatrix} x_c \\ y_c \end{pmatrix} - \frac{z_c}{\ddot{z}_c + g}\begin{pmatrix} \ddot{x}_c \\ \ddot{y}_c \end{pmatrix} + \frac{1}{m(\ddot{z}_c + g)}\begin{pmatrix} -\dot{l}_y \\ \dot{l}_x \end{pmatrix}, \tag{6}$$

where $g = 9.81$ m·s^{-2} is the gravity acceleration.

B. Task prediction models

In the following, we define a variety of tasks that will either be directly regulated in the MPC cost function, or used to constrain the robot motion. These include configuration-dependent tasks, that we denote as kinematic tasks, as well as the centroidal angular momentum task, and the ZMP task. For each of these, we are interested in finding a relationship affine in $\dot{\boldsymbol{\nu}}$. In the following, we denote with superscript $(\cdot)^k$ the related quantity at time t_k.

1) Kinematic tasks: Consider a generic configuration-dependent task $\boldsymbol{r}(\boldsymbol{q})$ and the corresponding differential kinematics as $\ddot{\boldsymbol{r}}(\boldsymbol{q}) = \boldsymbol{J}_r(\boldsymbol{q})\dot{\boldsymbol{\nu}} + \dot{\boldsymbol{J}}_r(\boldsymbol{q},\boldsymbol{\nu})\boldsymbol{\nu}$. In order to predict the evolution of the tasks over the control horizon, we linearize around the auxiliary trajectory $\bar{\boldsymbol{q}}(t), \bar{\boldsymbol{\nu}}(t)$, $t \in [t_k, t_{k+C}]$, defined as the MPC solution[1] computed at time t_{k-1}, and integrate using the linear time-varying prediction model

$$\begin{pmatrix} \boldsymbol{r}^{i+1} \\ \dot{\boldsymbol{r}}^{i+1} \end{pmatrix} = \begin{pmatrix} \boldsymbol{I} & \delta_t \boldsymbol{I} \\ \boldsymbol{0} & \boldsymbol{I} \end{pmatrix} \begin{pmatrix} \boldsymbol{r}^{i} \\ \dot{\boldsymbol{r}}^{i} \end{pmatrix} + \begin{pmatrix} \delta_t^2 \boldsymbol{I}/2 \\ \delta_t \boldsymbol{I} \end{pmatrix} \ddot{\boldsymbol{r}}^i, \tag{7}$$

with $\ddot{\boldsymbol{r}}^i = \boldsymbol{J}_r(\bar{\boldsymbol{q}}^i)\dot{\boldsymbol{\nu}}^i + \dot{\boldsymbol{J}}_r(\bar{\boldsymbol{q}}^i, \bar{\boldsymbol{\nu}}^i)\bar{\boldsymbol{\nu}}^i$, for $i = k, \dots, k+C-1$. By propagating (7) over the control horizon from an initial condition $(\boldsymbol{r}^k, \dot{\boldsymbol{r}}^k)$, we can express the predicted task position $\boldsymbol{R}^{k+1} = (\boldsymbol{r}^{k+1}, \dots, \boldsymbol{r}^{k+C})$ as:

$$\boldsymbol{R}^{k+1} = \boldsymbol{\Psi}_r \ddot{\boldsymbol{R}}^k + \boldsymbol{\psi}_r^k, \tag{8}$$

where $\ddot{\boldsymbol{R}}^k = (\ddot{\boldsymbol{r}}^k, \dots, \ddot{\boldsymbol{r}}^{k+C-1})$ is the vector of task accelerations depending on the predicted inputs, $\boldsymbol{\Psi}_r$ turns out to be a lower triangular strictly positive matrix, and $\boldsymbol{\psi}_r^k$ denotes the forward propagation of the current state.

The considered kinematic tasks include any relevant robot link, such as the left/right foot pose $\boldsymbol{p}_{lf}$, $\boldsymbol{p}_{rf}$, the torso orientation $\boldsymbol{\vartheta}_t$, as well as possibly the hands or the orientation of the head, and the joint configuration $\boldsymbol{q}_j$. Additionally, we leverage the same formulation to predict the CoM position $\boldsymbol{p}_c$, which will be used to control the CoM height z_c and to express the ZMP position (see Sect. III-B.3).

2) Angular momentum: In addition to the kinematic tasks, we are interested in regulating the centroidal angular momentum of the system. In this case, we use a first-order model, as in the second row of (7):

$$\boldsymbol{l}^{i+1} = \boldsymbol{l}^i + \delta_t \dot{\boldsymbol{l}}^i, \tag{9}$$

with $\dot{\boldsymbol{l}}^i = \boldsymbol{A}_l(\bar{\boldsymbol{q}}^i)\dot{\boldsymbol{\nu}}^i + \dot{\boldsymbol{A}}_l(\bar{\boldsymbol{q}}^i, \bar{\boldsymbol{\nu}}^i)\bar{\boldsymbol{\nu}}^i$, for $i = k, \dots, k+C-1$.

3) Zero Moment Point: By setting a constant CoM height z_c in (6), the relationship between the ZMP position on the ground plane and centroidal quantities becomes linear:

$$\begin{pmatrix} x_z^i \\ y_z^i \end{pmatrix} = \begin{pmatrix} x_c^i \\ y_c^i \end{pmatrix} - \frac{1}{\eta^2}\begin{pmatrix} \ddot{x}_c^i \\ \ddot{y}_c^i \end{pmatrix} + \frac{1}{mg}\begin{pmatrix} -\dot{l}_y^i \\ \dot{l}_x^i \end{pmatrix}, \tag{10}$$

for $i = k, \dots, k+C-1$, where $\eta = \sqrt{g/z_c}$. Defining $\ddot{\boldsymbol{X}}_c^k = (\ddot{x}_c^k, \dots, \ddot{x}_c^{k+C-1})$, $\dot{\boldsymbol{L}}_y^k = (\dot{l}_y^k, \dots, \dot{l}_y^{k+C-1})$ and $\boldsymbol{X}_z^k = (x_z^k, \dots, x_z^{k+C-1})$ equation 10 becomes

$$\boldsymbol{X}_z^k = \boldsymbol{X}_c^k + -\frac{1}{\eta^2}\ddot{\boldsymbol{X}}_c^k - \frac{1}{mg}\dot{\boldsymbol{L}}_y^k.$$

Using (8), the CoM trajectory $\boldsymbol{X}_c^k$ can be related to the task accelerations using its prediction matrix $\boldsymbol{\Psi}$, leading to

$$\boldsymbol{X}_z^k = \boldsymbol{M}\ddot{\boldsymbol{X}}_c^k + \boldsymbol{M}_{l_y}\dot{\boldsymbol{L}}_y^k + \boldsymbol{\mu}_x^k, \tag{11}$$

where $\boldsymbol{M}_{l_y} = -\frac{1}{mg}\boldsymbol{I}$ and $\boldsymbol{\mu}_x^k = (x_c^k, x_c^k + \delta_t \dot{x}_c^k, \dots, x_c^k + \delta_t(C-1)\dot{x}_c^k)$. Matrix $\boldsymbol{M}$ has a particular structure which will be used in the feasibility analysis. Let Ψ_{ij} be the (i,j) entry of $\boldsymbol{\Psi}$, then $\boldsymbol{M}$ can be written as

$$\boldsymbol{M} = -\frac{1}{\eta^2}\boldsymbol{I} + \begin{pmatrix} 0 & 0 & \dots & 0 \\ \Psi_{11} & 0 & \ddots & \\ \vdots & \ddots & \ddots & \vdots \\ \Psi_{C-1,1} & \dots & \Psi_{C-1,C-1} & 0 \end{pmatrix}.$$

Being $\boldsymbol{M}$ by construction lower triangular, with a strictly negative diagonal and positive off-diagonal terms, $-\boldsymbol{M}$ is a non-singular M-matrix [19]. In (11), accelerations $\dot{\boldsymbol{\nu}}$ appear explicitly using (5) evaluated on the auxiliary trajectory.

[1] At the start, when no previous solution is available, the algorithm is warmstarted using an initial guess.

IV. Constraints

In this section, we detail the different constraints that are enforced in the MPC.

A. ZMP constraint

Aside from the kinematic condition (2), in order to ensure contact stability it is also necessary that the ZMP remains strictly[2] inside the support polygon, i.e., the convex hull of contact surfaces. We employ a conservative approximation of the polygon in double support, which we refer to as *moving box* [20]. The moving box is a rectangular region of fixed shape and size, which corresponds with the footprint during single support phases, and shifts between consecutive footprints during double support phases. If the orientation of the box is constant over the horizon[3], the ZMP constraints are decoupled along x and y and can be written as

$$\begin{pmatrix} x_z^{\min,i} \\ y_z^{\min,i} \end{pmatrix} \leq \begin{pmatrix} x_z^{i} \\ y_z^{i} \end{pmatrix} \leq \begin{pmatrix} x_z^{\max,i} \\ y_z^{\max,i} \end{pmatrix}, \; i = k \dots, k+C-1.$$

For convenience, we express the constraints in matrix form separately for the x, y components. Recalling (11), the ZMP constraint for the x coordinate is then

$$\boldsymbol{X}_z^{\min,k} \leq \boldsymbol{M}\ddot{\boldsymbol{X}}_c^k + \boldsymbol{M}_{l_y}\dot{\boldsymbol{L}}_y^k + \boldsymbol{\mu}_x^k \leq \boldsymbol{X}_z^{\max,k}, \tag{12}$$

with $\boldsymbol{X}_z^{\min,k}$ and $\boldsymbol{X}_z^{\max,k}$ collecting the bounds over the control horizon.

B. Stability constraint

Controlling the ZMP is necessary for the stability of contacts, but it is not enough to produce a stable gait since the CoM may still diverge [21]. We therefore use the CoM stability constraint introduced in [13].

The unstable component of the CoM dynamics can be isolated by performing the change of coordinates

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

often referred to as the Divergent Component of Motion (DCM). To ensure bounded ZMP-CoM trajectories we enforce a terminal constraint on (x_u, y_u). For x_u it writes as

$$x_u^{k+C} = x_c^{k+C} + \frac{\dot{x}_c^{k+C}}{\eta} = \tilde{x}_u^{k+C}$$

where $\tilde{x}_u^{k+C}$ is defined as

$$\tilde{x}_u^{k+C} = \eta \int_{t_{k+C}}^{\infty} e^{-\eta(\tau - t_{k+C})} x_z(\tau) d\tau. \tag{14}$$

In order to compute this term, we assume to have an available approximation $\tilde{x}_z(t)$ in $[t_{k+C}, t_{k+H}]$ of the future ZMP trajectory derived from the footstep plan, and that that $\tilde{x}_z(t) = \tilde{x}_z^{k+H}$ for $t \geq t_{k+H}$. In the terminology of [13], this corresponds to using an *anticipative tail*.

Here we also assume that $\dot{l}_x$ and $\dot{l}_y$ are equal to zero for $t \geq t_{k+C}$. In principle, one could conjecture a future trajectory for $\boldsymbol{l}$, but since it will be regulated to zero, this turns out to be a sensible choice. We can now split the integral over two time intervals and compute

$$\eta \int_{t_{k+H}}^{\infty} e^{-\eta(\tau - t_{k+C})} \tilde{x}_z^{k+H} d\tau = \underbrace{e^{-\eta(t_{k+H} - t_{k+C})} \tilde{x}_z^{k+H}}_{\tilde{x}_z^{\infty}}.$$

which plugged in (14), gives

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

Finally, assuming that $\tilde{x}_z$ is a piecewise-constant anticipative tail, we have

$$\tilde{x}_u^{k+C} = (1 - e^{-\eta\delta_t}) \sum_{i=C}^{H-1} e^{-i\eta\delta_t} \tilde{x}_z^{k+i} + \tilde{x}_z^{\infty}.$$

The stability constraint can then be written as a function of the CoM accelerations $\ddot{\boldsymbol{X}}_c$ as:

$$x_u^{k+C} = \boldsymbol{\Phi}_u \ddot{\boldsymbol{X}}_c^k + \varphi_x^k = \tilde{x}_u^{k+C}, \tag{15}$$

with $\boldsymbol{\Phi}_u$ being the last row the matrix $\boldsymbol{\Psi}_u$ which describes the prediction of x_u over the control horizon and $\varphi_x^k = x_c^k + \left(\frac{1}{\eta} + C\delta_t\right) \dot{x}_c^k$. Matrix $\boldsymbol{\Psi}_u$ is obtained by applying the change of coordinates (13) to the CoM prediction model of the form 8. In (15), accelerations $\dot{\boldsymbol{\nu}}$ appear explicitly using (5) evaluated on the auxiliary trajectory. Similar considerations can be made for the y coordinate.

C. Joint limits

In order to make the predicted motion kinematically feasible, we impose position limits on the joints over the prediction horizon:

$$\boldsymbol{q}_j^{\min} \leq \boldsymbol{q}_j^i \leq \boldsymbol{q}_j^{\max} \quad i = k+1, \dots, k+C. \tag{16}$$

D. Contact constraint

The contact constraint (2) is enforced over the control horizon by imposing zero velocity to the support foot:

$$\dot{\boldsymbol{p}}_{sf}^i = \boldsymbol{0}, \quad i = k+1, \dots, k+C. \tag{17}$$

V. JIS-MPC

This section will define the QP to be solved at each iteration by JIS-MPC.

The cost function includes terms aimed at executing the kinematic tasks, minimizing the angular momentum $\boldsymbol{l}$, tracking the feasibility region center $\tilde{\boldsymbol{f}}$, keeping a constant CoM height $z_{c,d}$, and regulating the upper-body joints to their initial configuration $\boldsymbol{q}_j^0$ for resolving redundancy.

The feasibility-related terms $\boldsymbol{f}$ and $\tilde{\boldsymbol{f}}$ will be defined in Sect. VI, where we will conduct a centroidal feasibility analysis of the proposed MPC scheme. Even though it might seem like their inclusion could lead to a recursive definition, this is not the case. In fact, $\tilde{\boldsymbol{f}}$ does not depend on the cost function because the feasibility of any QP is fully determined solely by its constraints.

[2] Normally, strict inequalities cannot be enforced, but the available area for the constraint is reduced from the full contact surface.

[3] The constant orientation is automatically verified if the robot walks in a straight line, but can also be obtained when the robot rotates, by slightly shrinking the box with respect to the footprint.

Let the decision variables be $\boldsymbol{w}^k = (\dot{\boldsymbol{\nu}}^k, ..., \dot{\boldsymbol{\nu}}^{k+C-1})$. At time t_k, JIS-MPC solves the following QP:

$$\left\{ \begin{array}{l} \min\limits_{\boldsymbol{w}^k} \sum\limits_{i=k+1}^{k+C} \left\|\boldsymbol{r}^i - \boldsymbol{r}_d^i\right\|_{\boldsymbol{W}_r}^2 + \left\|\dot{\boldsymbol{q}}_j^i\right\|_{\boldsymbol{W}_{\dot{q}_j}}^2 + \left\|\boldsymbol{l}^i\right\|_{\boldsymbol{W}_l}^2 + \left\|\boldsymbol{f}^i - \tilde{\boldsymbol{f}}^i\right\|_{\boldsymbol{W}_f}^2 \\ \quad + \sum\limits_{i=k}^{k+C-1} \left\|\dot{\boldsymbol{\nu}}^i\right\|_{\boldsymbol{W}_{\dot{\nu}}}^2 \\ \text{subject to:} \\ \bullet \ \text{stability constraint (15) for } x \text{ and } y, \\ \bullet \ \text{ZMP constraints (12) for } x \text{ and } y, \\ \bullet \ \text{joint limits (16),} \\ \bullet \ \text{contact constraint (17),} \end{array} \right.$$

where $\boldsymbol{r}$ indicates the stack of kinematic tasks, i.e., the torso orientation, feet pose, joint positions, and CoM height, obtained following (8).

After computing the solution, the first optimal acceleration $\dot{\boldsymbol{\nu}}^{*,k}$ is selected from the optimal sequence $\boldsymbol{w}^{*,k}$, and the joint commands $\boldsymbol{u}^{*,k}$ are applied to the system. Then, the optimal sequence is integrated to generate an auxiliary trajectory $\bar{\boldsymbol{q}}(t)$, $\bar{\boldsymbol{\nu}}(t)$ for the next iteration.

VI. Feasibility Analysis

We are interested in studying the feasibility of the JIS-MPC optimization problem. To do so, we consider its minimal set of constraints, i.e., the stability constraint (15) and ZMP constraint (12). This will allow us to characterize a feasibility region that is defined at the level of the centroidal dynamics, thus independent on the joint configuration. While this is a simplification, it still provides significant value. In fact, even though we are operating at the joint level, the centroidal state turns out to be the major contributor to determining the feasibility of the QP. The reason for this is given by the fact that the robot is very redundant, and kinematic limitations (including contact constraints) are unlikely to limit centroidal motions in typical locomotion scenarios. In fact, these constraints are actively enforced in the MPC and we have empirically verified that in our tests the success of the QP was accurately predicted by the centroidal feasibility region.

Moreover, in the present analysis we neglect the angular momentum contribution to the ZMP constraint, setting $\boldsymbol{M}_{l_x} = \boldsymbol{M}_{l_y} = 0$. Indeed, although this term is important for performance and robustness, it would be necessary to impose proper bounds on it to carry out a similar analysis, which we reserve for future work.

A. Centroidal feasibility region

Let $f_x^k = \varphi_x^k - \boldsymbol{\Phi}_u \boldsymbol{M}^{-1} \boldsymbol{\mu}_x^k$, a quantity[4] which depends on the current CoM position and velocity $(x_c^k, \dot{x}_c^k)$, and similarly define f_y^k. By combining the stability (15) and ZMP constraints (12), it is possible to find lower/upper bounds for $\boldsymbol{f}^k = (f_x^k, f_y^k)$ that ensure the MPC being feasible.

Proposition 1: If matrix $\boldsymbol{A}_c(\bar{\boldsymbol{q}}^{k+i})$, for $i = 0, \ldots, C-1$ is nonsingular[5], then, JIS-MPC is feasible at time t_k if and only if the current state $(\boldsymbol{q}^k, \boldsymbol{\nu}^k)$ is such that $\boldsymbol{f}^k \in \mathcal{F}^k$, where

$$\mathcal{F}^k = \left\{ (f_x, f_y) : \begin{array}{c} f_x^{\min,k} \leq f_x \leq f_x^{\max,k} \\ f_y^{\min,k} \leq f_y \leq f_y^{\max,k} \end{array} \right\},$$

with feasibility region bounds

$$\begin{array}{l} f_x^{\min,k} = \tilde{x}_u^{k+C} - \boldsymbol{\Phi}_u \boldsymbol{M}^{-1} \boldsymbol{X}_z^{\min,k} \\ f_x^{\max,k} = \tilde{x}_u^{k+C} - \boldsymbol{\Phi}_u \boldsymbol{M}^{-1} \boldsymbol{X}_z^{\max,k}, \end{array} \tag{18}$$

and equivalent definitions for the y coordinate.

Proof. The assumption on the CoM Jacobians ensures sufficient redundancy such that there exist some joint commands that realize any given CoM acceleration profile $\ddot{\boldsymbol{X}}_c^k$. Then, we need to prove that if the current state is such that $\boldsymbol{f}^k \in \mathcal{F}^k$ there exists a solution that satisfies both stability and ZMP constraints, and that there are no such trajectories if $\boldsymbol{f}^k$ is outside of the feasibility region.

To prove necessity, assume that $\boldsymbol{f}^k \notin \mathcal{F}^k$. For the x coordinate, either $f_x^k > f_x^{\max,k}$ or $f_x^k < f_x^{\min,k}$. Let us first account for the former, that is

$$f_x^k = f_x^{\max,k} + \Delta f_x, \quad \Delta f_x > 0, \tag{19}$$

and assume that there exists an acceleration profile $\ddot{\boldsymbol{X}}_c^*$ for which both the stability and the ZMP constraints (15) and (12) are satisfied. As a consequence of the latter, it must be that

$$\boldsymbol{M}\ddot{\boldsymbol{X}}_c^* + \boldsymbol{\mu}_x^k = \boldsymbol{X}_z^{\max,k} - \Delta \boldsymbol{z}$$

with $\Delta \boldsymbol{z}$ a vector of nonnegative components. Left-multiplying by $\boldsymbol{M}^{-1}$ and rearranging we obtain

$$\ddot{\boldsymbol{X}}_c^* - \boldsymbol{M}^{-1}(\boldsymbol{X}_z^{\max,k} - \boldsymbol{\mu}_x^k) = -\boldsymbol{M}^{-1} \Delta \boldsymbol{z}. \tag{20}$$

Let

$$\begin{array}{l} \ddot{\boldsymbol{X}}_c^{\min,k} = \boldsymbol{M}^{-1} \left(\boldsymbol{X}_z^{\min,k} - \boldsymbol{\mu}_x^k \right) \\ \ddot{\boldsymbol{X}}_c^{\max,k} = \boldsymbol{M}^{-1} \left(\boldsymbol{X}_z^{\max,k} - \boldsymbol{\mu}_x^k \right), \end{array} \tag{21}$$

then (20) becomes

$$\ddot{\boldsymbol{X}}_c^* - \ddot{\boldsymbol{X}}_c^{\max,k} = -\boldsymbol{M}^{-1} \Delta \boldsymbol{z}.$$

Left-multiplying by $\boldsymbol{\Phi}_u$, we obtain

$$\boldsymbol{\Phi}_u (\ddot{\boldsymbol{X}}_c^* - \ddot{\boldsymbol{X}}_c^{\max,k}) = -\boldsymbol{\Phi}_u \boldsymbol{M}^{-1} \Delta \boldsymbol{z} \geq 0. \tag{22}$$

This inequality holds because $\boldsymbol{\Phi}_u$ is a positive matrix by definition, $-\boldsymbol{M}^{-1}$ is a positive matrix, since $-\boldsymbol{M}$ is a nonsingular M-matrix, and $\Delta \boldsymbol{z}$ is nonnegative by definition. We now prove that actually the left side of (22) is strictly negative, leading to contradiction. In fact, imposing the stability constraint (15) with $\ddot{\boldsymbol{X}}_c^k = \ddot{\boldsymbol{X}}_c^*$ and recalling the definition of f_x^k, (18), and (19), one obtains:

$$\boldsymbol{\Phi}_u \ddot{\boldsymbol{X}}_c^* + \underbrace{f_x^{\max,k} + \boldsymbol{\Phi}_u \boldsymbol{M}^{-1} \boldsymbol{\mu}_x^k}_{\tilde{x}_u^{k+C} - \boldsymbol{\Phi}_u \boldsymbol{M}^{-1}(\boldsymbol{X}_z^{\max,k} - \boldsymbol{\mu}_x^k)} + \Delta f_x = \tilde{x}_u^{k+C}.$$

[4] This quantity represents the predicted free evolution (i.e., with $x_z = 0$) of the unstable component of the dynamics x_u at the end of the prediction horizon.

[5] The Jacobians being full rank is actually a requirement only for sufficiency. Also, empirical results showed this to be the case, thanks to the high degree of redundancy of the system.

which combined with (21) leads to

$$\begin{aligned}\boldsymbol{\Phi}_u\ddot{\boldsymbol{X}}_c^* + \tilde{x}_u^{k+C} - \boldsymbol{\Phi}_u\ddot{\boldsymbol{X}}_c^{\max,k} + \Delta f_x &= \tilde{x}_u^{k+C}\\ \boldsymbol{\Phi}_u(\ddot{\boldsymbol{X}}_c^* - \ddot{\boldsymbol{X}}_c^{\max,k}) &= -\Delta f_x < 0.\end{aligned}$$

Thus, it is impossible that $\ddot{\boldsymbol{X}}_c^*$ satisfies both constraints if $f_x^k > f_x^{\max,k}$. The same reasoning can be repeated for the case $f_x^k < f_x^{\min,k}$ and for the y coordinate, proving necessity.

We now prove that if $f_x^k \in [f_x^{\min,k}, f_x^{\max,k}]$, a feasible solution $\ddot{\boldsymbol{X}}_c^*$ exists. In fact,

$$\exists\, \alpha_x \in [0,1] : f_x^k = \alpha_x f_x^{\min,k} + (1-\alpha_x) f_x^{\max,k}.$$

Thanks to the convexity of the constraints, it is then possible to prove that the acceleration profile

$$\ddot{\boldsymbol{X}}_c^* = \alpha_x \ddot{\boldsymbol{X}}_c^{\min,k} + (1-\alpha_x)\ddot{\boldsymbol{X}}_c^{\max,k} \tag{23}$$

satisfies the stability and ZMP constraints along x. The same reasoning can be applied to the y coordinate. ■

B. Tracking of the feasibility region center

Having defined the feasibility region $\mathcal{F}^k$, it is possible to compute its center $\tilde{\boldsymbol{f}}^k$, which depends only on the predicted ZMP bounds. This can be used by the MPC to favor trajectories that will keep the system far from the bounds of the feasibility region to improve robustness. In fact, assuming that the time horizon over which the ZMP bounds are available is $P \geq C + H$, the feasibility region center over the control horizon (for $i = k+1, \ldots, k+C$) can be computed as

$$\tilde{f}_x^i = \frac{f_x^{\min,i} + f_x^{\max,i}}{2} = \tilde{x}_u^{i+C} + \frac{1}{2}\boldsymbol{\Phi}_u \boldsymbol{M}^{-1}\big(\boldsymbol{X}_z^{\min,i} + \boldsymbol{X}_z^{\max,i}\big)$$

for the x coordinate, and similarly for y.

VII. Results

We performed a series of dynamic simulations on a Kawada HRP-4 robot using the DART simulation environment. For comparison, each simulation is replicated using the same settings on IS-MPC with fixed footsteps. Joint limits are enforced based on the specifications contained in the URDF description.

Since the simulations are dynamic, CoM position and velocities are affected by disturbances, such as imperfect contacts and external forces. CoM measurements are passed through a Kalman filter, whose model is a double integrator with CoM acceleration as input. Interestingly, although IS-MPC does not work if the measures are not filtered, JIS-MPC does work correctly, and the filter is only added in order to make the comparison fair.

We first show a typical gait followed by more challenging ones. Moreover, we test the robustness of the method by applying impulsive disturbances of increasing intensity, and report detailed comparative results. Animations of all the simulations are also reported in the accompanying video[6].

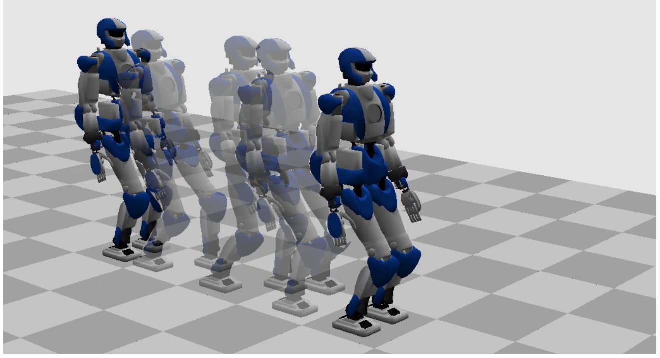

Fig. 1. Simulation snapshots for the standard gait obtained with average speed 0.3 m/s and step duration 1 s.

Settings: The number of actuated joints is $n = 20$, having locked some redundant joints in the upper body to reduce the number of variables. The same parameters and weights are used in *all* simulations: $\delta_t = 0.1$ s, $T_C = 1$ s, $T_H = 2$ s, $\boldsymbol{W}_{\dot{\nu}} = 10^{-4}\boldsymbol{I}$, $\boldsymbol{W}_f = 0.5\boldsymbol{I}$, $\boldsymbol{W}_l = \mathrm{diag}\{10^{-4}, 10^{-4}, 1\}$, $\boldsymbol{W}_r = \mathrm{diag}\{\boldsymbol{W}_{\vartheta_t}, \boldsymbol{W}_{lf}, \boldsymbol{W}_{rf}, W_{z_c}, \boldsymbol{W}_{q_j}\}$ with $\boldsymbol{W}_{\vartheta_t} = 10^{-2}\boldsymbol{I}$, $\boldsymbol{W}_{lf} = \boldsymbol{W}_{rf} = 500\boldsymbol{I}$, $W_{z_c} = 1$; for the joints, we only regulate the chest pitch and yaw, the left/right shoulder pitch and roll, elbow pitch and hip yaw, so $\boldsymbol{W}_{q_j} = 10^{-3}\mathrm{diag}\{10, 5, 10, 50, 5, 5, 10, 50, 5, 5, \mathbf{0}_{10\times 1}\}$, and $\boldsymbol{W}_{\dot{q}_j} = 10^{-2}\boldsymbol{W}_{q_j}$. The size of the ZMP constraint is 0.1 m along x and y.

Reference trajectories: Footstep plans are generated by setting the duration of a step and calculating the position around a straight line path in such a way to achieve a desired average speed. The desired swing trajectory follows a linear profile for x and y and a parabolic profile along z, with maximum height equal to 0.06 m. Since we only report simulations in which the robot walks along the x axis, the torso reference angle is always assumed to be equal to the starting orientation. Unless stated otherwise, the desired CoM height is $z_{c,d} = 0.77$ m.

Implementation details: The code is accessible via an open-source repository[7]. All kinematic and dynamic quantities of the model used for control are obtained using the Pinocchio library. Optimization problems are solved using `hpipm`. The average computation time in a typical simulation is around 5 ms, including the QP preparation, on an AMD Ryzen 9 5900X (4.8 GHz) with 16 GB DDR4 3600 MHz running Ubuntu 22.04 LTS, making the scheme fully compatible with real-time requirements.

A. Standard gait

The standard gait is generated with an average speed of 0.3 m/s and step duration of 1 s, divided in 0.7 s for single support and 0.3 s for double support. Snapshots of the simulation are reported in Fig. 1. The robot is able to execute the footstep plan, with both the lower and upper body being involved in the gait. Note that the arm swing motion in opposition to the foot swing only arises due to the minimization of the vertical angular momentum l_z.

[6] Available at `https://youtu.be/Fa6iy3mUcBY`

[7] `https://github.com/DIAG-Robotics-Lab/joint_level_ismpc`

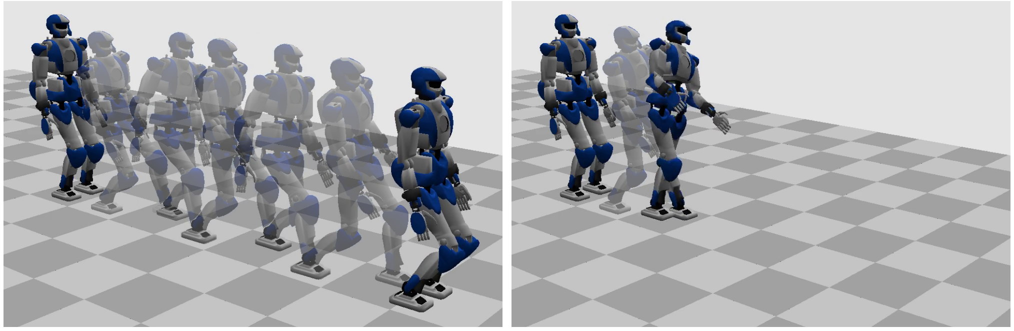

Fig. 2. Simulation snapshots for the long-step gait, with average speed 0.5 m/s and step duration 1 s: the motion is executed successfully when joint position limits are enforced (left), while it fails if these are removed due to the knee reaching a singular configuration (right).

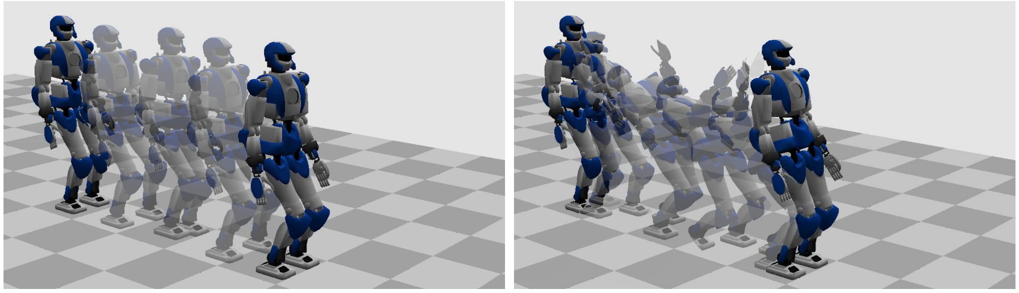

Fig. 3. Simulation snapshots for the short-step gait, with average speed 0.3 m/s and step duration 0.6 s: the motion is natural when tracking the center of the feasibility region (left), but becomes very unnatural when tracking the center of the ZMP constraint (right).

Standard IS-MPC can generate a stable gait with these settings, however the arm swinging motion is not generated by itself and must be superimposed if one wishes to obtain a natural-looking gait.

B. Challenging gaits

We now report simulations for more challenging gaits, in which standard IS-MPC is unable to produce a stable solution due to the accelerations becoming too high and violating the assumptions of the LIP model.

1) Long-step gait: first is a gait with average speed of 0.5 m/s and same step duration as the standard gait, shown in Fig. 2. In this case, the desired CoM height has been lowered to $z_{c,d} = 0.76$ m to make the task kinematically feasible. This results in longer leg swings, and demonstrates how the introduction of joint position limits is necessary to perform movements that require the full kinematic capabilities of the robot. In fact, the very same settings fail if joint limits are removed, because the knee joint reaches a singularity.

2) Short-step gait: the second is a gait with average speed of 0.3 m/s and step duration of 0.6 s, divided in 0.4 s for single support and 0.2 s for double support. The resulting motion, shown in Fig. 3, is characterized by higher-frequency steps, requiring large vertical accelerations of the swing foot.

Interestingly, these settings provide a testbed for evaluating the usefulness of the tracking of the feasibility region center. In fact, if this is removed and replaced with a classic term in which the ZMP tracks the center of the moving box, then the generated movements become very unnatural, as the robot tilts its torso backwards to gain angular momentum for moving the ZMP. On the other hand, we have not experienced such issues when tracking the center of the feasibility region, where the angular momentum is not involved. Indeed, one could tune the cost function weights, to limit this behavior when tracking the ZMP, by penalizing upper-body movements; however, such tuning would be very specific to the task and ultimately detrimental to robustness, as upper-body movements are shown to be crucial in rejecting disturbances.

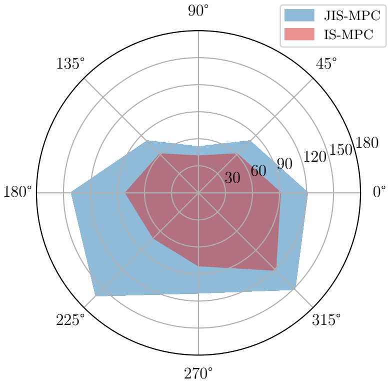

Fig. 4. Comparison of robustness to impulsive disturbances of duration 0.1 s applied from various directions with intensity of $[30, 180]$ N. Angles represent the direction from which the force is coming from. Note how the proposed method is able to tolerate larger disturbances, indicated by the colored region, especially when applied from the 3rd quadrant.

C. Robustness to impulsive disturbances

Finally, we performed a simulation campaign in order to evaluate the ability to react to an impulsive force. In this case, the footstep plan is generated with an average speed of 0.2 m/s and same step duration as the standard gait. Disturbances from 12 different directions, equally distributed

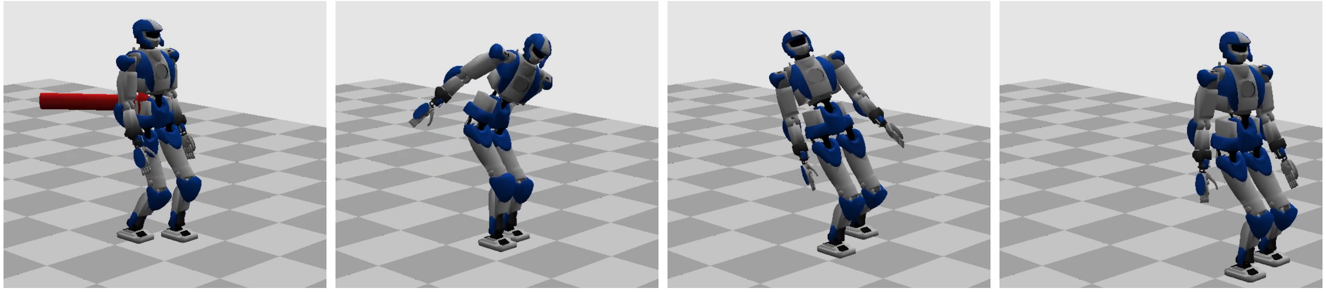

Fig. 5. Snapshots of the robot motion after being pushed with a force of 140 N from the 225° direction (red arrow).

around the robot, are applied at the torso. The magnitude of the force starts at 30 N and goes up to 180 N in increments of 10 N. All pushes are applied at $t = 3$ s for a duration of 0.1 s. Again, we compare with standard IS-MPC: we report the maximum force that each method was able to sustain for each direction in the radial plot of Fig. 4, which shows the clear advantage of JIS-MPC. In fact, in accordance with the results of [11], the introduction of the angular momentum allows the robot to better react to disturbances, as it translates to a more accurate ZMP prediction. This is particularly evident for pushes coming from the rear-right direction (3rd quadrant of Fig. 4). An instance of these is shown in Fig. 5 where, upon receiving the push, the robot is initiating a swing with the right foot.

VIII. Conclusions

We proposed a whole-body MPC based on a centroidal dynamics + kinematics formulation, enforcing a stability constraint. The proposed scheme offers significant performance and robustness improvements over an MPC-based on a simplified model. Our feasibility study (see Sect. VI) will let us utilize the same feasibility-driven results that we developed for standard IS-MPC, with minimal adaptation.

Future work will include:

- exploiting the feasibility information for online footstep adaptation;
- extension to 3D motions of the CoM in a world of stairs [20], where direct access to joint variables is crucial for clearing taller steps;
- extending the feasibility study to consider the angular momentum contribution, as well as the presence of other constraints (e.g., joint limits and contact constraints);
- extending the work to torque-controlled robots.

References

[1] V. Tsounis, M. Alge, J. Lee, F. Farshidian, and M. Hutter, "Deepgait: Planning and control of quadrupedal gaits using deep reinforcement learning," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3699–3706, 2020.

[2] P.-B. Wieber, R. Tedrake, and S. Kuindersma, "Modeling and control of legged robots," in *Springer Handbook of Robotics*, B. Siciliano and O. Khatib, Eds. Springer, 2016, pp. 1203–1234.

[3] S. Feng, X. Xinjilefu, W. Huang, and C. G. Atkeson, "3D walking based on online optimization," in *2013 IEEE-RAS Int. Conf. on Humanoid Robots*, 2013, pp. 21–27.

[4] G. Romualdi, S. Dafarra, G. L'Erario, I. Sorrentino, S. Traversaro, and D. Pucci, "Online non-linear centroidal MPC for humanoid robot locomotion with step adjustment," in *2022 IEEE Int. Conf. on Robotics and Automation*, 2022, p. 10412–10419.

[5] L. Rossini, E. Mingo Hoffman, S. H. Bang, L. Sentis, and N. G. Tsagarakis, "A real-time approach for humanoid robot walking including dynamic obstacles avoidance," in *2023 IEEE-RAS Int. Conf. on Humanoid Robots*, 2023, pp. 1–8.

[6] J. Koenemann, A. Del Prete, Y. Tassa, E. Todorov, O. Stasse, M. Bennewitz, and N. Mansard, "Whole-body model-predictive control applied to the HRP-2 humanoid," in *2015 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2015, pp. 3346–3351.

[7] P. M. Wensing, M. Posa, Y. Hu, A. Escande, N. Mansard, and A. D. Prete, "Optimization-based control for dynamic legged robots," *IEEE Trans. on Robotics*, vol. 40, pp. 43–63, 2024.

[8] E. Dantec, M. Naveau, P. Fernbach, N. Villa, G. Saurel, O. Stasse, M. Taix, and N. Mansard, "Whole-body model predictive control for biped locomotion on a torque-controlled humanoid robot," in *2022 IEEE-RAS Int. Conf. on Humanoid Robots*, 2022, p. 638–644.

[9] M. Djeha, P. Gergondet, and A. Kheddar, "Robust task-space quadratic programming for kinematic-controlled robots," *IEEE Trans. on Robotics*, 2023.

[10] H. Li, R. J. Frei, and P. M. Wensing, "Model hierarchy predictive control of robotic systems," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3373–3380, 2021.

[11] C. Khazoom and S. Kim, "Humanoid arm motion planning for improved disturbance recovery using model hierarchy predictive control," in *2022 IEEE Int. Conf. on Robotics and Automation*, 2022, p. 6607–6613.

[12] H. Dai, A. Valenzuela, and R. Tedrake, "Whole-body motion planning with centroidal dynamics and full kinematics," in *2014 IEEE-RAS Int. Conf. on Humanoid Robots*, 2014, p. 295–302.

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

[14] F. M. Smaldone, N. Scianca, L. Lanari, and G. Oriolo, "Feasibility-driven step timing adaptation for robust MPC-based gait generation in humanoids," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 1582–1589, 2021.

[15] A. S. Habib, F. M. Smaldone, N. Scianca, L. Lanari, and G. Oriolo, "Handling non-convex constraints in MPC-based humanoid gait generation," in *2022 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2022, pp. 13 167–13 173.

[16] M. Cipriano, M. R. Maximo, N. Scianca, L. Lanari, and G. Oriolo, "Feasibility-aware plan adaptation in humanoid gait generation," in *2023 IEEE-RAS Int. Conf. on Humanoid Robots*, 2023, pp. 1–8.

[17] D. E. Orin, A. Goswami, and S.-H. Lee, "Centroidal dynamics of a humanoid robot," *Autonomous Robots*, vol. 35, no. 2–3, p. 161–176, 2013.

[18] S. Caron, Q.-C. Pham, and Y. Nakamura, "Stability of surface contacts for humanoid robots: Closed-form formulae of the contact wrench cone for rectangular support areas," in *2015 IEEE Int. Conf. on Robotics and Automation*, 2015, p. 5107–5112.

[19] R. Plemmons, "M-matrix characterizations. I—nonsingular M-matrices," *Linear Algebra and its Applications*, vol. 18, no. 2, pp. 175–188, 1977.

[20] M. Cipriano, P. Ferrari, N. Scianca, L. Lanari, and G. Oriolo, "Humanoid motion generation in a world of stairs," *Robotics and Autonomous Systems*, vol. 168, p. 104495, 2023.

[21] T. Takenaka, T. Matsumoto, and 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.