2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)
Birmingham, UK, November 15-17, 2017, pp. 178-183.

Humanoid Gait Generation for Walk-To Locomotion using Single-Stage MPC

Ahmed Aboudonia, Nicola Scianca, Daniele De Simone, Leonardo Lanari, Giuseppe Oriolo

***Abstract*— We consider the problem of gait generation for a humanoid robot that must walk to an assigned Cartesian goal. As a first solution, we consider a rather straightforward adaptation of our previous work: an external block produces high-level velocities, which are then tracked by a double-stage intrinsically stable MPC scheme where the orientation of the footsteps is chosen before determining their location and the CoM trajectory. The second solution, which represents the main contribution of the paper, is conceptually different: no high-level velocity is generated, and footstep orientations are chosen at the same time of the other decision variables in a single-stage MPC. This is made possible by carefully redesigning the motion constraints so as to preserve linearity. Preliminary results on a simulated NAO confirm that the single-stage method outperforms the conventional double-stage scheme.**

I. Introduction

One of the most challenging problems in the control of humanoids is generating locomotion gaits. The main requirement is obviously that the robot maintains dynamic balance while walking. One way to guarantee this is to move the Zero Moment Point (ZMP, the point where the horizontal component of the moment of the ground reaction forces becomes zero) in such a way that it is always inside the support polygon of the robot. Many gait generation schemes enforce this ZMP condition by computing a suitable trajectory for the robot Center of Mass (CoM). Due to the complexity of humanoid dynamics, simplified models are invariably used to relate the evolution of the CoM to that of the ZMP. A popular choice is the second-order linear system known as the Linear Inverted Pendulum (LIP) [1]. Once a CoM trajectory is generated, kinematic control provides joint commands that drive the robot along it.

In the literature, walking gaits have been mainly designed for the case of persistent locomotion, in which the robot must track a non-decaying velocity, e.g., see [2]. Here, we consider instead the problem of walk-to locomotion, in which the humanoid has to reach an assigned goal in the workspace and stop there. This is a relevant case in applications, e.g., for a humanoid robot that must move to various locations of its environment to execute certain tasks. Some footstep planning algorithms exist for this situation, such as [3], [4]; however, these planners may not be able to run in real time or lack robustness in the presence of perturbations or changes in the environments. A related approach is presented in [5], where inverse optimal control is applied to walk-to locomotion, focusing however more on the replication of human behavior than on actual gait generation.

Model Predictive Control is a powerful tool for generating walking gaits. First, it makes possible to enforce constraints, such as the fact that the ZMP must remain within the support polygon. Second, it provides considerable robustness to perturbations. For real-time computation, it is desirable to derive a quadratic programming problem, where the constraints must be linear in the decision variables. However, the ZMP constraints are found to be nonlinear with respect to the footstep orientations. In [6] this problem is circumvented by choosing the latter before solving for the other variables. Another approach is taken in [7], where the nonlinear optimization problem is solved by using Mixed-Integer quadratic programming.

In this paper, we consider two MPC-based approaches for generating walk-to gaits. The first is a rather straightforward adaptation of the intrinsically stable MPC framework proposed in [8]: an external block produces high-level velocities, which are then tracked by a double-stage MPC scheme where the orientation of the footsteps is chosen before determining their location and the CoM trajectory. The second, which represents the main contribution of the paper, does not require high-level reference velocities, because the Cartesian regulation action is directly embedded in the cost function. In addition, footstep orientations are chosen at the same time of the other decision variables in a single-stage MPC. This is made possible by carefully redesigning the motion constraints so as to preserve linearity.

The paper is organized as follows. In Sect. II we define the problem and outline the double-stage and single-stage approaches, which are described in detail in Sect. III and Sect. IV, respectively. Comparative simulations on a NAO humanoid robot are presented in Sect. V, while Sect. VI mentions some extensions we are currently considering.

II. Problem and Approaches

Consider a situation in which a humanoid robot must reach a given (planar) goal in the workspace. To achieve this, it is necessary to generate a walking gait that leads the robot to the goal and stop its motion there (*walk-to*). Throughout the paper, we assume that the humanoid knows the relative position of the goal with respect to itself, an information that can be reconstructed using for instance an on-board camera.

We consider two MPC-based approaches. In the first (Fig. 1, top), an external block produces high-level velocities, which are then tracked by a double-stage MPC scheme where the orientation of the footsteps is chosen before determining

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. This work is supported by the EU H2020 project COMANOID.

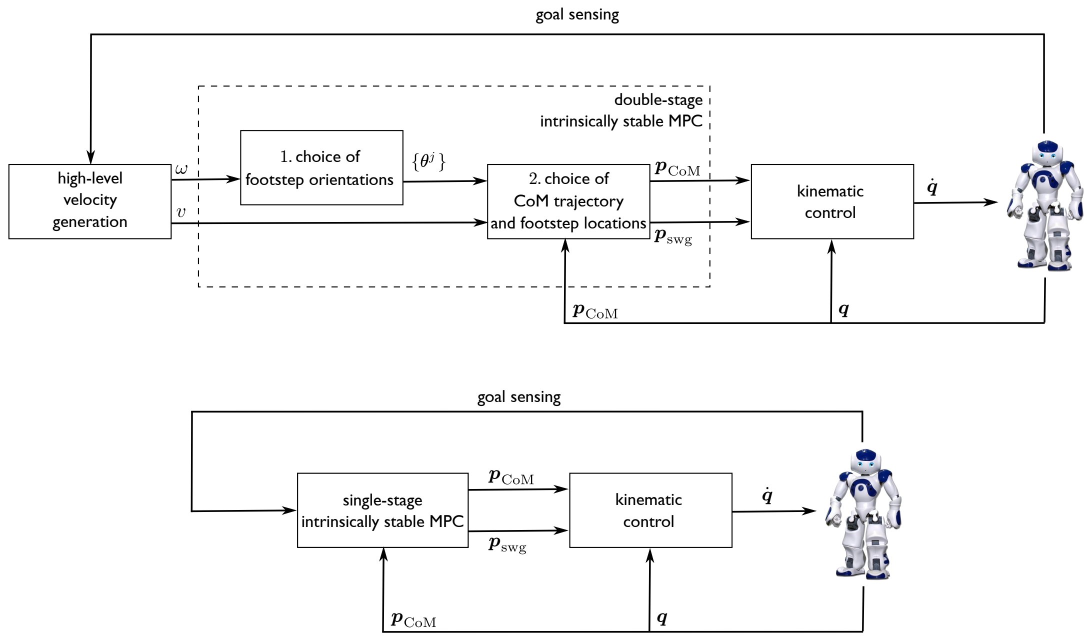


Fig. 1. Possible approaches for walk-to gait generation using MPC. *Top*: an adaptation of the intrinsically stable gait generation framework proposed in [8], using an external block for high-level velocity generation and a double-stage MPC. *Bottom*: the single-stage MPC approach proposed in this paper.

their location and the CoM trajectory. The second (Fig. 1, bottom), which represents the main contribution of the paper, is conceptually different: no high-level velocity is generated and, moreover, footstep orientations are chosen at the same time of the other decision variables.

In the next sections we discuss these two approaches, called respectively *double-stage* and *single-stage MPC*.

III. Gait Generation via Double-Stage MPC

A description is now given of the individual blocks of the approach based on double-stage MPC (Fig. 1, top). We use a local frame attached the robot, having as origin the current projection of the robot CoM on the ground. The x (*sagittal*) axis is aligned with the support foot while the y (*coronal*) axis points in the orthogonal direction to it.

A. High-Level Velocity Generation

This block generates high-level velocity commands to be used as reference signals by the MPC scheme.

As in [9], [10], we will adopt a unicycle as template model for high-level velocity generation. In particular, consider a unicycle placed at the origin of the current robot frame and aligned with the sagittal axis. Control velocities are produced by the following Cartesian regulator [11]:

$$\begin{aligned} v &= k_1 \, x_g \\ \omega &= k_2 \cdot \mathrm{atan2}(y_g, x_g), \end{aligned}$$

where $k_1, k_2 > 0$ and (x_g, y_g) are the Cartesian coordinates of the goal in the local frame (assumed to be known, see Section II). Note that the driving velocity v is proportional to the projection of the Cartesian error on the x axis, while the angular velocity ω is proportional to the pointing error.

B. Double-stage MPC

All MPC computations are performed in the local robot frame at the current sampling instant t_k.

1) Choice of footstep orientations: To maintain linearity of the constraints in our MPC formulation, the orientation of the footsteps must be chosen before determining their location and the CoM trajectory [6].

Denote by T_h the MPC prediction horizon, and assume that the robot performs steps of constant duration T_s: the number of footsteps within T_h is then $M = \mathrm{ceil}(T_h/T_s)$. The orientations $\theta^1, \ldots, \theta^M$ of these footsteps in the current robot frame are chosen on the basis of the angular velocity ω (assumed to be constant[1] within T_h) by minimizing the quadratic cost function

$$\sum_{j=1}^{M} \left(\frac{\theta^j - \theta^{j-1}}{T_s} - \omega \right)^2$$

subject to the linear constraint $|\theta^j - \theta^{j-1}| \leq \theta_{\max}$, where $\theta_{\max}$ is the maximum acceptable change of orientation between two consecutive footsteps.

2) Choice of CoM trajectory and footstep locations: Once footstep orientations have been chosen, the gait must be completed by the CoM trajectory and the footstep locations.

The motion model consists of two identical, decoupled and dynamically extended LIPs, one along the sagittal axis and the other along the coronal axis. For example, the sagittal

[1]This assumption, also made for v, is justified by the short T_h which our intrinsically stable MPC scheme can accommodate (more on this later).

model is

$$\begin{pmatrix} \dot{x}_{\rm c} \\ \ddot{x}_{\rm c} \\ \dot{x}_{\rm z} \end{pmatrix} = \begin{pmatrix} 0 & 1 & 0 \\ \eta^2 & 0 & -\eta^2 \\ 0 & 0 & 0 \end{pmatrix} \begin{pmatrix} x_{\rm c} \\ \dot{x}_{\rm c} \\ x_{\rm z} \end{pmatrix} + \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \dot{x}_{\rm z},$$

where $x_{\rm c}$ and $x_{\rm z}$ are respectively the CoM and ZMP coordinates, and $\eta = \sqrt{g/h}$, with h the constant CoM height.

The MPC uses sampling intervals of duration δ over the prediction horizon $T_h = N\,\delta$. At time t_k, the decision variables are the piecewise-constant ZMP velocities during the next N sampling intervals, denoted by $\dot{x}_{\rm z}^{k+i}$, $\dot{y}_{\rm z}^{k+i}$, for $i = 1, \ldots, N$, and the location of the next M footsteps, denoted by $(x_{\rm f}^j, y_{\rm f}^j)$, for $j = 1, \ldots, M$.

The cost function to be minimized is

$$\sum_{i=1}^{N} \left((\dot{x}_{\rm z}^{k+i})^2 + k_x(\dot{x}_{\rm c}^{k+i} - v\cos(i\omega\delta))^2 + (\dot{y}_{\rm z}^{k+i})^2 + k_y(\dot{y}_{\rm c}^{k+i} - v\sin(i\omega\delta))^2 \right), \tag{1}$$

which includes two terms penalizing deviations from the high-level reference velocities.

Footstep locations do not appear in the cost function and influence the QP problem through the linear constraints to which the problem is subject, i.e. the ZMP constraint for maintaining balance, the CoM stability constraint, and the kinematic feasibility constraint on the footstep location.

The first constraint imposes that the ZMP must always lie inside the robot support polygon. In single support, this corresponds to defining a rectangle centered at $(x_{\rm f}^j, y_{\rm f}^j)$, oriented as θ^j, and having dimensions $x_{\rm z}^{\rm dim}, y_{\rm z}^{\rm dim}$, where the ZMP must fall[2]:

$$R_j^T \begin{pmatrix} \delta \sum_{l=k+1}^{k+i} \dot{x}_{\rm z}^l - x_{\rm f}^j \\ \delta \sum_{l=k+1}^{k+i} \dot{y}_{\rm z}^l - y_{\rm f}^j \end{pmatrix} \leq \frac{1}{2} \begin{pmatrix} x_{\rm z}^{\rm dim} \\ y_{\rm z}^{\rm dim} \end{pmatrix} - R_j^T \begin{pmatrix} x_{\rm z}^k \\ y_{\rm z}^k \end{pmatrix}, \tag{2}$$

where R_j is the rotation matrix associated to angle θ^j.

To guarantee the boundedness of the CoM trajectory regardless of the duration of the prediction horizon $T_{\rm h}$, we enforce the stability constraint

$$\frac{1}{\eta} \frac{1 - e^{\delta\eta}}{1 - e^{N\delta\eta}} \sum_{i=1}^{N} e^{i\delta\eta} \dot{x}_{\rm z}^{k+i} = x_{\rm c}^k + \frac{\dot{x}_{\rm c}^k}{\eta} - x_{\rm z}^k, \tag{3}$$

where we must set $x_{\rm c}^k = 0$ in view of the use of a local frame. For an interpretation of this constraint, see [8]; comparative simulations have shown that its adoption allows to reduce the prediction horizon $T_{\rm h}$ without jeopardizing gait stability.

The last constraint ensures kinematic feasibility of the footstep locations:

$$R_{j-1}^T \begin{pmatrix} x_{\rm f}^j - x_{\rm f}^{j-1} \\ y_{\rm f}^j - y_{\rm f}^{j-1} \end{pmatrix} \leq \pm \begin{pmatrix} 0 \\ \ell \end{pmatrix} + \frac{1}{2} \begin{pmatrix} x_{\rm f}^{\rm dim} \\ y_{\rm f}^{\rm dim} \end{pmatrix}, \tag{4}$$

where $x_{\rm f}^{\rm dim}$ and $y_{\rm f}^{\rm dim}$ are the dimensions of the rectangular feasible area, and ℓ is its displacement from the previous footstep in the coronal direction. The signs in the rhs alternate for the two feet.

[2]For compactness, in the following we only specify the right-hand-side of inequality constraints.

IV. Gait Generation via Single-Stage MPC

We now come to the description of the method for walk-to gait generation proposed in this paper, i.e., the single-stage intrinsically stable MPC framework of Fig. 1, bottom.

Note that no high-level velocity generation block is present; the Cartesian regulation action is directly embedded in the new MPC cost function. The first stage of the previous scheme has also been eliminated by choosing the footstep orientations at the same time of the other decision variables and properly redefining all constraints to preserve linearity, with the exception of the stability constraint (3) which remains linear and is therefore unchanged.

The reference frame and motion model used in single-stage MPC are exactly the same of double-stage MPC. In the following, we discuss the new cost function and the redefinition of the constraints.

A. Cost Function

The cost function for single-stage MPC is

$$J = \alpha_z \sum_{i=1}^{N} \left((\dot{x}_{\rm z}^{k+i})^2 + (\dot{y}_{\rm z}^{k+i})^2 \right) + \alpha_g \sum_{i=1}^{N} \left((x_g - x_{\rm c}^{k+i})^2 + (y_g - y_{\rm c}^{k+i})^2 \right) + \alpha_\theta \sum_{j=1}^{M} (\theta_g^j - \theta^j)^2 + \alpha_v \sum_{i=1}^{N} (\dot{x}_{\rm c}^{k+i} - v_{\rm ref})^2,$$

where α_z, α_g, α_θ, α_v are positive weights.

The first term is obviously the control effort and is also present in (1). The second term accounts for the positioning error with respect to the goal. The role of the third term is to force the robot to align its footsteps to the line of sight to the goal; this is obtained by defining θ_g^j as

$$\theta_g^j = j\,\lambda\,T_s \cdot \mathrm{atan2}\,(y_g, x_g)\,,$$

where $\lambda > 0$ is a control gain.

In the fourth term, $v_{\rm ref}$ represent a reference speed along the sagittal axis, defined to be constant up to a certain distance to the goal, from where it decreases linearly to become zero at the goal. The effect of this term is to avoid large velocities away from the goal and, on the other hand, excessively slow motion in its vicinity.

B. Constraints

1) Single-support ZMP constraint: The single-support ZMP constraint (2) becomes obviously nonlinear if the orientation θ^j of the footstep is still a decision variable. To avoid this problem, we redefine the constraint so that it becomes independent on the foot orientation.

In particolar, consider the construction in Fig. 2. For illustration, assume that the prediction horizon only includes one footstep ($M = 1$) and $x_{\rm z}^{\rm dim} = y_{\rm z}^{\rm dim}$ (square footprint). The blue square represents the current footstep, while the red squares are two different placements of the predicted footstep (same location but different orientations). The green square,

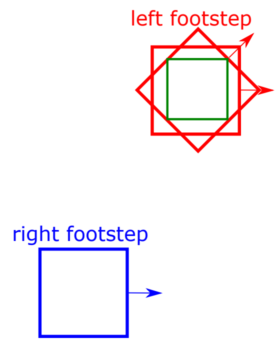


Fig. 2. Redefining the ZMP constraint during single support.

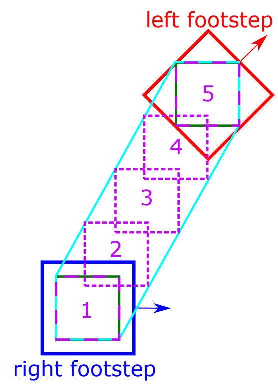


Fig. 3. Redefining the ZMP constraint during double support.

which has the same orientation as the current footstep but size reduced by a factor of $\sqrt{2}$, is always contained in the red squares, irrespective of their orientation. Thus, if the ZMP is located inside the green square, it is certainly contained in the actual footprint, whatever its orientation.

In conclusion, the ZMP constraint during single support can be redefined for any value of M as

$$\begin{pmatrix} \delta \sum_{l=k+1}^{k+i} \dot{x}_z^l - x_f^j \\ \delta \sum_{l=k+1}^{k+i} \dot{y}_z^l - y_f^j \end{pmatrix} \leq \frac{1}{2} \begin{pmatrix} \tilde{x}_z^{\dim} \\ \tilde{y}_z^{\dim} \end{pmatrix} - \begin{pmatrix} x_z^k \\ y_z^k \end{pmatrix},$$

where $\tilde{x}_z^{\dim} = x_z^{\dim}/\sqrt{2}$ and $\tilde{y}_z^{\dim} = y_z^{\dim}/\sqrt{2}$.

The above procedure for preserving linearity obviously implies a small reduction of the ZMP constraint area with respect to the actual footprint. However, this effect is more than balanced by the overall increase in the area that becomes feasible for stepping thanks to the inclusion of the foosteps orientation in the main MPC formulation (see Fig. 4).

2) Double-support ZMP constraint: In a double support phase, part of the boundary of the support polygon is defined by a nonlinear function of the relative position of the two feet. For this reason, the ZMP constraint is usually ignored during double support. In the following, we take inspiration from [12] to redefine this constraint so that it becomes linear and can therefore be enforced.

Consider the construction shown in Fig. 3. As before, we consider the case $M = 1$ for illustration. The support polygon during double support is shown in cyan. The ZMP should move from the green square (the single-support ZMP constraint area redefined previously) within the right footstep to the green square within the left footstep. Suppose that this transition takes D sampling intervals ($D = 5$ in the figure). This leads to define D equispaced support squares, shown in purple in Fig. 3. During the n-th time-step ($n = 1, \dots, 5$), the support square n is activated. By doing so, the ZMP is always contained inside the original double support polygon while preserving the linearity of the ZMP constraints.

According to the above discussion, the ZMP constraint during double support is redefined as

$$\begin{pmatrix} \delta \sum_{l=k+1}^{k+i} \dot{x}_z^l - \frac{n}{F} x_f^{j-1} - \frac{F-n}{F} x_f^j \\ \delta \sum_{l=k+1}^{k+i} \dot{y}_z^l - \frac{n}{F} y_f^{j-1} - \frac{F-n}{F} y_f^j \end{pmatrix} \leq \frac{1}{2} \begin{pmatrix} \tilde{x}_z^{\dim} \\ \tilde{y}_z^{\dim} \end{pmatrix} - \begin{pmatrix} x_z^k \\ y_z^k \end{pmatrix},$$

with $n = 1, \dots, 5$ and $F = D + 1$.

3) Kinematic feasibility constraint: As the ZMP constraint during single support, the kinematic feasibility constraint (4) becomes nonlinear if the orientations of the footsteps have not been chosen yet. To circumvent this problem, we redefine also this constraint appropriately.

Figure 4 shows two different predictions (same location, different orientations) for a right footstep and the corresponding feasible areas (solid line) for placing the next left footstep. Note that both the location and the orientation of these areas depends on the orientation of the right footstep. To remove this dependency, a reduced feasible area (dashed line) is defined in each original area. This reduced region, whose orientation is fixed, is then translated based on the orientation of the right footstep. By forcing the ZMP to be inside the union of all translated regions, we guarantee that it is also inside the union of the original feasibility areas. This is a linear constraint which can be written as

$$\begin{pmatrix} x_f^{j+1} - x_f^j - \ell\,\theta^j \\ y_f^{j+1} - y_f^j \end{pmatrix} \leq \begin{pmatrix} \tilde{x}_f^{\dim}/2 \\ \ell + \tilde{y}_f^{\dim}/2 \end{pmatrix},$$

with $\tilde{x}_f^{\dim}, \tilde{y}_f^{\dim}$ the dimensions of the reduced feasible area.

Note that the above construction is valid for the case in which two footsteps must to be placed within the prediction horizon ($M = 2$). In principle, it can be extended to the case of multiple predicted footsteps, but the reduced feasible area shrinks quickly, so that it is only practical for small M. Once again, the fact that our intrinsically stable MPC can work with a small T_h is beneficial also under this viewpoint.

4) Maximum foot rotation constraint: Finally, we must directly add the linear constraint

$$|\theta^j - \theta^{j-1}| \leq \theta_{\max},$$

which was previously enforced only in the first stage of MPC (see Sect. III-B.1).

V. Comparative Simulations

We now present some simulation results for a NAO humanoid robot in V-REP. Gait generation is performed in real-time with a control rate of 100 Hz. We have used a sampling interval $\delta = 0.01$ s, a step duration $T_s = 0.3$ s (0.2 s of single support and 0.1 s of double support) and a prediction horizon $T_h = 0.6$ s, i.e. one gait period. Other parameters

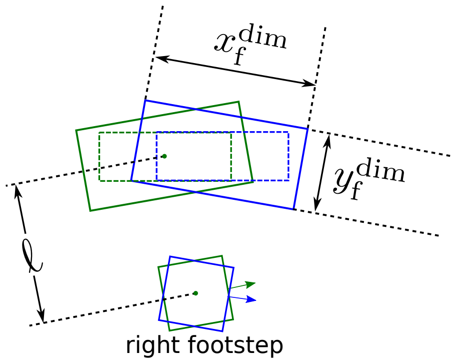
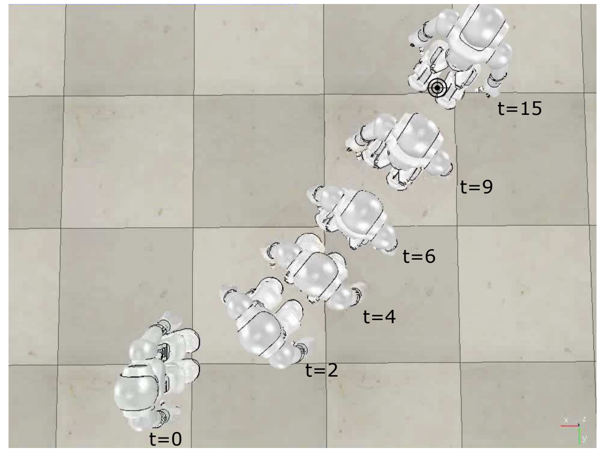


Fig. 4. Redefining the kinematic feasibility constraint for footstep locations.



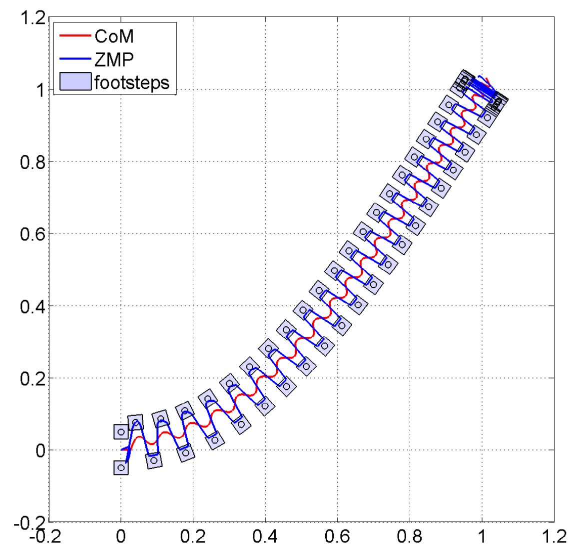


Fig. 5. Simulation 1. NAO successfully walks to the assigned goal using single-stage MPC (top); the robot CoM, ZMP and footsteps along the generated motion (bottom).

are $h_{\text{com}} = 0.26$ m, $x_z^{\text{dim}} = y_z^{\text{dim}} = 0.04$ m, $x_f^{\text{dim}} = 0.1$ m, $y_f^{\text{dim}} = 0.05$ m, $\ell = 0.125$ m, and $\theta_{\max} = \pi/16$. Finally, we have set $k_1 = 0.25$, $k_2 = 1$, and $k_x = k_y = 10$ for double-stage MPC; and $\alpha_z = \alpha_v = 1$, $\alpha_g = 10$, $\alpha_\theta = 2$, $\lambda = 1$, $\tilde{x}_f^{\text{dim}} = 0.08$ m and $\tilde{y}_f^{\text{dim}} = 0.03$ m for single-stage MPC. QP problems were solved using qpOASES.

The first simulation, shown in Fig. 5, simply confirms that the proposed single-stage MPC can effectively produce a walk-to gait. Note how the robot first aligns with the goal and then walks in a straight line, naturally decelerating in the vicinity of the goal and finally stopping there.

The second and third simulations are presented to highlight the superiority of the proposed single-stage MPC with respect to the double-stage technique. In particular, the second simulation (Fig. 6) refers to a situation where an impulsive push is applied to the robot during a single support phase. Double-stage MPC, in which the ZMP constraints are not enforced in double support, produces a solution that brings the ZMP outside the support polygon. Single-stage MPC is instead able to produce a feasible solution because it includes ZMP constraints in double support. Note also how the ZMP profile in the single-stage solution penetrates the footprints more than in the double-stage solution, due to the reduction of the ZMP constraint area in the former case.

In view of the results of the second simulation, we have modified the double-stage MPC by including the same double support ZMP constraint of single-stage MPC. Simulations of this modified double-stage MPC have however revealed another intrinsic drawback of the approach, namely the rigidity introduced by the fact that the footstep orientations are chosen before the other decision variables. For example, Fig. 7 refers to another situation where an impulsive push is applied to the robot during a single support phase. After the push, double-stage MPC cannot find a solution that satisfies the double support ZMP constraint because the footstep orientations are fixed. Instead, single-stage MPC reacts to the push by adjusting the foot orientations. The robot can then successfully complete the walk-to motion.

Simulation clips are shown in the accompanying video.

VI. Conclusions

In this paper we have introduced a single-stage MPC framework for generating walk-to gaits for humanoid robots. The novel formulation does not require an external block for generating reference velocities, and allows the MPC to compute simultaneously footsteps locations and orientations while preserving the linearity of the constraints. To increase the robustness of the scheme, we have enforced the ZMP constraint also in the double support phase. Results on a simulated NAO confirm the superiority of the single-stage method over the conventional double-stage scheme.

We are currently working towards an experimental validation the proposed approach. We also intend to extend the proposed method to take into account the presence of workspace obstacles, following the ideas in [13].

References

[1] S. Kajita, H. Hirukawa, K. Harada, and K. Yokoi, *Introduction to Humanoid Robotics*. Springer, 2014.

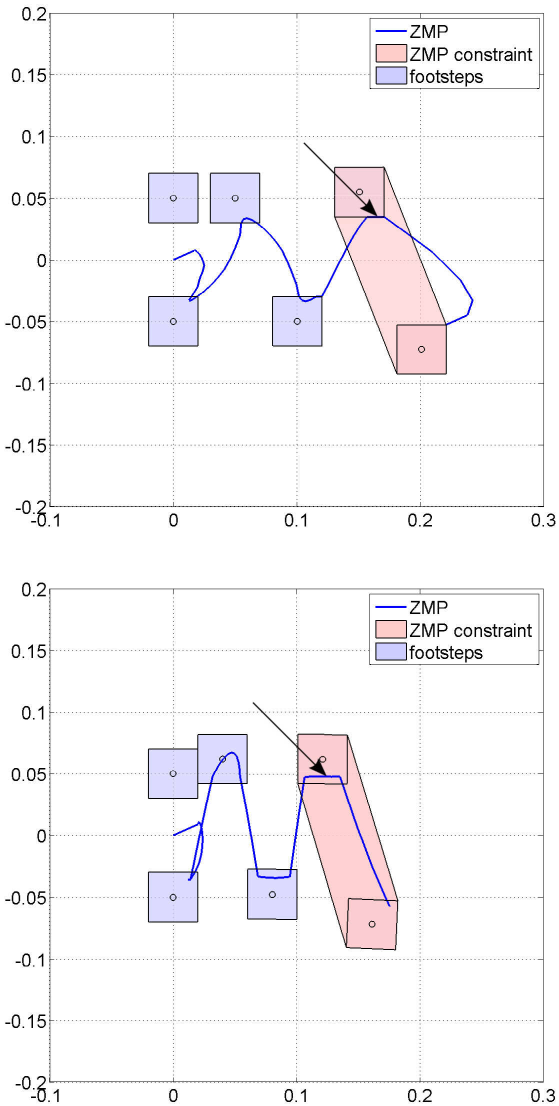


Fig. 6. Simulation 2. Comparison between double-stage MPC (top) and single-stage MPC (bottom) when a push (represented by an arrow) is applied to the robot. Note how the ZMP constraint in double support is violated with double-stage MPC.

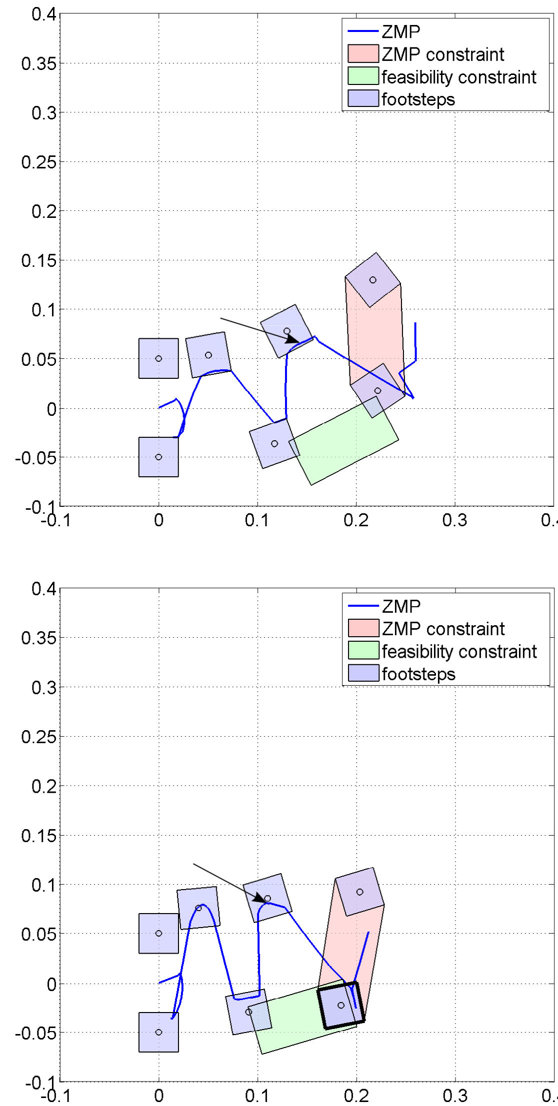


Fig. 7. Simulation 3. Another comparison between double-stage MPC (top) and single-stage MPC (bottom) when a push (represented by an arrow) is applied to the robot. Both schemes now enforce the ZMP constraint in double support; however, double-stage MPC cannot find a feasible solution because the footstep orientations are fixed. Single-stage MPC easily accommodates the push by slightly rotating the highlighted footstep.

[2] 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.

[3] R. Deits and R. Tedrake, "Footstep planning on uneven terrain with mixed-integer convex optimization," in *2014 14th IEEE-RAS Int. Conf. on Humanoid Robots*, 2014, pp. 279–286.

[4] P. Karkowski, S. Oßwald, and M. Bennewitz, "Real-time footstep planning in 3d environments," in *2016 IEEE-RAS 16th Int. Conf. on Humanoid Robots*, 2016, pp. 69–74.

[5] K. Mombaur, A. Truong, and J.-P. Laumond, "From human to humanoid locomotion – an inverse optimal control approach," *Autonomous Robots*, vol. 28, pp. 369–383, 2010.

[6] A. Herdt, N. Perrin, and P. B. Wieber, "Walking without thinking about it," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 190–195.

[7] R. J. Griffin and A. Leonessa, "Model predictive control for dynamic footstep adjustment using the divergent component of motion," in *2016 IEEE Int. Conf. on Robotics and Automation*, 2016, pp. 1763–1768.

[8] N. Scianca, M. Cognetti, D. De Simone, L. Lanari, and G. Oriolo, "Intrinsically stable MPC for humanoid gait generation," in *16th IEEE-RAS Int. Conf. on Humanoid Robots*, 2016, pp. 101–108.

[9] M. Cognetti, D. De Simone, L. Lanari, and G. Oriolo, "Real-time planning and execution of evasive motions for a humanoid robot," in *2016 IEEE Int. Conf. on Robotics and Automation*, 2016, pp. 4200–4206.

[10] M. Cognetti, D. De Simone, F. Patota, N. Scianca, L. Lanari, and G. Oriolo, "Real-time pursuit-evasion with humanoid robots," in *2017 IEEE Int. Conf. on Robotics and Automation*, 2017, pp. 4090–4095.

[11] B. Siciliano, L. Sciavicco, L. Villani, and G. Oriolo, *Robotics: Modelling, Planning and Control*. Springer, 2009.

[12] D. Dimitrov, A. Paolillo, and P.-B. Wieber, "Walking motion generation with online foot position adaptation based on ℓ_1- and ℓ_∞-norm penalty formulations," in *2011 IEEE Int. Conf. on Robotics and Automation*, 2011, pp. 3523–3529.

[13] D. De Simone, N. Scianca, P. Ferrari, L. Lanari, and G. Oriolo, "MPC-based humanoid pursuit-evasion in the presence of obstacles," in *2017 IEEE/RSJ Int. Conf. on Intelligent Robots and Systems*, 2017.