Model Predictive Control of Biped Walking with Bounded Uncertainties

Nahuel Villa, Pierre-Brice Wieber

HAL Id: hal-01649580

https://hal.inria.fr/hal-01649580

Submitted on 27 Nov 2017

Model Predictive Control of Biped Walking with Bounded Uncertainties

Nahuel A. Villa and Pierre-Brice Wieber

Abstract— **A biped walking controller for humanoid robots has to handle together hard constraints, dynamic environments, and uncertainties. Model Predictive Control (MPC) is a suitable and widely used control method to handle the first two issues. Uncertainties on the robot imply a non-zero tracking error when trying to follow a reference motion. A standard solution for this issue is to use tighter constraints by introducing some hand tuned safety margins, for the reference motion generation to ensure that the actual robot motion will satisfy all constraints even in presence of the tracking error. In this article, we find bounds for the tracking error and we show how such safety margins can be precisely computed from the tracking error bounds. Also, a tracking control gain is proposed to reduce the restrictiveness introduced with the safety margins. MPC with these considerations ensure the correct operation of the biped robot under a given degree of uncertainties when it is implemented in open-loop. Nevertheless, the straightforward way to implement an MPC closed-loop scheme fails. We discuss the reasons for this failure and propose a robust closed-loop MPC scheme.**

I. INTRODUCTION

Walking depends on contact forces between the feet and the ground. The unilateral nature of this interaction (feet can only push on the ground) limits the motion that a legged robot can realize and plays a crucial role in its stability. When walking on a flat ground, this corresponds to having the Center of Pressure (CoP) stay within the support polygon [25]. Model Predictive Control (MPC) is one of few suitable methods to handle such constraints [17], and has been used therefore extensively for the control of legged robots. This allows generating walking motions online with automatic footstep placement [14], taking into account visual feedback [8], avoiding collisions in a crowd [3], undertaking physical collaborations with humans [1].

In a complex system such as a humanoid robot, sources of uncertainties abound: noisy sensors, imperfect actuators, unmodeled dynamics, inaccurate models of the environment, *etc*. These sources of uncertainty can always be reduced with more precise (and more expensive) hardware or we can try to estimate them, but only up to a point: there always remains a certain amount of uncertainty. And as a result, there always remains a certain amount of tracking error when trying to follow a reference motion [5], [12], [9], [25], [15]. In order to satisfy all constraints in the presence of such tracking error, the usual approach is to introduce hand tuned safety margins [24]. The first contribution of this paper is to show how such safety margins can be computed precisely based on the tracking error produced by a given set of uncertainties.

Univ. Grenoble Alpes, Inria, CNRS, Grenoble INP*, LJK, 38000 Grenoble, France `nahuel.villa@inria.fr;` `pierre-brice.wieber@inria.fr`.

The reference motion is usually generated with an MPC scheme independently from the actual robot motion [11]. We will refer to this approach as Open Loop MPC (OL–MPC). There are situations, however, where it is necessary to generate a reference motion taking into account the actual state of the robot [13], such as when there is a strong perturbation. A classical approach is to generate the reference motion starting directly from the actual state of the robot. We will refer to this approach as Closed Loop MPC (CL–MPC). Unfortunately, we are going to see that this approach can fail. In an approach discussed in [18], the initial state of the reference motion is actually chosen to take into account the actual state of the robot but also ensure explicitly robustness. We will refer to this approach as Robust Closed Loop MPC (RCL–MPC).

Robust control is an approach to controller design that explicitly deals with uncertainty. One possible aim is to reduce the probability of failure [1], [6], [7]. But here, we aim at ensuring the correct operation of the system for any uncertainty between certain bounds.

Section II briefly introduces the dynamics of walking. Section III proposes an analysis of tracking error under bounded uncertainties, and introduces the key concept of Robust Positive Invariant (RPI) set. Section IV discusses how to optimize tracking control gains in order to minimize the necessary safety margins. Section V introduces different combinations of Model Predictive Control (MPC) scheme and higher-frequency tracking control that can be found in the literature. Section VI demonstrates with simple simulations how some of these schemes can fail while others succeed.

II. POINT MASS MODEL WITH BOUNDED UNCERTAINTIES

Let us consider the Center of Mass (CoM) c of a biped robot walking on a flat horizontal ground. From Newton and Euler equations of the whole robot, we obtain the Center of pressure (CoP) [25] as:

$$p^{x,y} = c^{x,y} - \frac{mc^z \ddot{c}^{x,y} + S\dot{L}^{x,y}}{m(\ddot{c}^z + g^z)} - v^{x,y}, \quad (1)$$

with L the angular momentum, g^z the vertical acceleration due to gravity, m the robot mass, $S = \begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}$ a rotation matrix and v represents a bounded uncertainty:

$$v \in W. \quad (2)$$

Because contact forces with the ground are unilateral (the robot can push but not pull on the ground), the CoP is bound

to lie in the support polygon $\mathcal{P}$:

$$p \in \mathcal{P}. \tag{3}$$

This dynamics is traditionally reformulated as follows in order to emphasize the linear relationship between the horizontal motion of the CoM and the CoP [9], [1], [24], [23], [5], [10], dropping coordinate indices ^{x} and ^{y}:

$$\ddot{c} = \omega^2(c - p - n - v), \tag{4}$$

where $\omega^2 = g^z/c^z$ and

$$n = \frac{\ddot{c}}{\omega^2} - \frac{mc^z\ddot{c} + S\dot{L}}{m(\ddot{c}^z + g^z)}. \tag{5}$$

We can bound the values of n as discussed in [4], [21]:

$$-n \in \mathcal{N}. \tag{6}$$

Assuming that p, n and v are constant over time intervals of length τ, typically 5(ms), we can obtain a linear time-invariant discrete-time system following a standard procedure [19]:

$$x^+ = Ax + B(p + n + v), \tag{7}$$

with

$$A = \begin{bmatrix} \cosh(\omega\tau) & \omega^{-1}\sinh(\omega\tau) \\ \omega\sinh(\omega\tau) & \cosh(\omega\tau) \end{bmatrix},$$
$$B = \begin{bmatrix} 1 - \cosh(\omega\tau) \\ -\omega\sinh(\omega\tau) \end{bmatrix},$$
$$x = \begin{bmatrix} c \\ \dot{c} \end{bmatrix} \in \mathcal{X}, \tag{8}$$

where x^+ is the successor state. Here, we introduce the state constraint $\mathcal{X}$ to take into account maximum leg length.

III. ROBUST POSITIVE INVARIANT SET

Let us consider a reference motion x_{ref} following the nominal, undisturbed linear dynamics

$$x_{ref}^+ = Ax_{ref} + Bp_{ref}. \tag{9}$$

Because of uncertainties v and model errors n in the dynamics (7), we have to consider a (linear) tracking control law (with compensation of n)

$$p = p_{ref} - n + K(x - x_{ref}) \tag{10}$$

to properly follow this reference motion. With this control law, the tracking error

$$\delta = x - x_{ref} \tag{11}$$

follows the closed-loop dynamics

$$\delta^+ \in (A + BK)\delta + BW. \tag{12}$$

Iterating this closed-loop dynamics N times, the tracking error becomes:

$$\delta^{+N} \in (A + BK)^N\delta + \bigoplus_{i=0}^{N-1}(A + BK)^i BW, \tag{13}$$

where $\oplus$ is the Minkowski sum[1]. If the gain K stabilizes the matrix $A + BK$ (eigenvalues inside the unit circle), then when $N \to \infty$ the first term vanishes:

$$(A + BK)^N\delta \to 0 \tag{14}$$

and $\delta^{+N} \to Z$, where

$$Z = \bigoplus_{i=0}^{\infty}(A + BK)^i BW. \tag{15}$$

When there is no uncertainty, $W = \{0\}$, the tracking error converges to the origin, $Z = \{0\}$, as desired. But if there are uncertainties, what we see here is that the tracking error δ will converge to a bigger set Z, which depends on W.

We can observe from (15) that

$$(A + BK)Z \oplus BW = Z, \tag{16}$$

what implies, following the closed-loop dynamics (12), that

$$\forall\delta \in Z, \quad \forall v \in W, \quad \delta^+ \in Z. \tag{17}$$

Once the tracking error reaches Z, it stays in Z. This set is Robust Positively Invariant (RPI) [20], and can serve as a bound on tracking error in steady-state. It can be shown that the set Z defined above is actually the *minimal* RPI (mRPI) set (in the sense of being a subset of any RPI set of (12)) [20], so it provides the smallest bound possible on tracking error. Naturally, when $\delta \in Z$, $K\delta \in KZ$. In that case, in order to satisfy the constraints (3) and (8) with the tracking error (11) and the control law (10), the reference motion must clearly satisfy

$$x_{ref} \in \mathcal{X} \ominus Z, \tag{18}$$
$$p_{ref} \in \mathcal{P} \ominus \mathcal{N} \ominus KZ, \tag{19}$$

where $\ominus$ is the Pontryagin difference[2]. Here, Z and KZ play respectively the role of state and control safety margins for the design of the reference motion x_{ref}.

IV. OPTIMAL TRACKING CONTROL GAINS

It can be interesting to use a tracking control gain K that minimizes the safety margins Z and KZ. In our case the CoP p is the most constrained element in the dynamics. Therefore, we choose to minimize more specifically

$$KZ = \bigoplus_{i=0}^{\infty} K(A + BK)^i BW. \tag{20}$$

Note that if

$$K(A + BK) = 0, \tag{21}$$

all terms with $i > 0$ vanish and the above sum becomes $KZ = KBW$. We show in the Appendix that this happens with

$$K = k \begin{bmatrix} 1 & \omega^{-1} \end{bmatrix} \quad \text{and} \quad k = \frac{e^{\omega\tau}}{e^{\omega\tau} - 1}. \tag{22}$$

[1]Given sets A and B, $A \oplus B := \{a + b \mid a \in A, b \in B\}$.
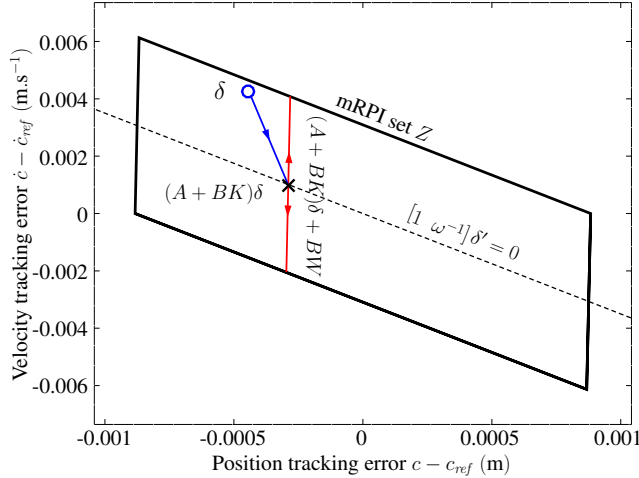[2]Given sets A and B, $A \ominus B := \{x \mid x + B \subseteq A\}$.

Fig. 1: With the choice of gain (22), the minimal Robust Positive Invariant (mRPI) set Z is a parallelogram. Any tracking error δ (blue circle) is driven by the closed-loop matrix $A + BK$ onto the line $[1 \quad \omega^{-1}]\delta' = 0$ (dashed line). An uncertainty is then added from the set BW (red segment).

We confirmed numerically by computing KZ for the entire range of stabilizing gains, that this choice of K actually minimizes the safety margin KZ.

Introducing the Capture Point (CP)

$$\xi = c + \omega^{-1}\dot{c} = \begin{bmatrix} 1 & \omega^{-1} \end{bmatrix} x, \tag{23}$$

we can observe from (21) and (22) that for any tracking error δ,

$$\begin{bmatrix} 1 & \omega^{-1} \end{bmatrix} (A + BK)\delta = 0, \tag{24}$$

so the CP tracking error is driven to 0 in one sampling period. Such a dead-beat control of the CP was already proposed in [9] for different reasons as uncertainties were not considered. It corresponds also to "the best regulator" proposed in [23]. The set Z corresponding to this choice of gain K, using parameters provided in Table I, can be seen in Fig.1, together with the typical behavior of the closed-loop dynamics (12) of the tracking error.

With this choice of gain, the control safety margin becomes

$$KZ = KBW = -e^{\omega\tau}W. \tag{25}$$

As a result, the maximal set of uncertainties such that $KZ \subseteq \mathcal{P}$, so that the constraint (19) is nonempty (Neglecting $\mathcal{N}$), is

$$W_{max} = -e^{-\omega\tau}\mathcal{P}. \tag{26}$$

For example assuming a source of uncertainties in the form of an external horizontal force f acting on the CoM of the robot,

$$v = -\frac{f}{m\omega^2}, \tag{27}$$

parameters in Table I would lead to a maximal force $f_{max} = 48(\text{N})$.

TABLE I: Numerical Values

Parameter	Symbol	Value	Unit
CoM Height	c^z	0.8	(m)
Robot Mass	m	58	(kg)
Gravity Acceleration	g	9.81	(m.s^{-2})
Natural Frequency	ω	3.5	(s^{-1})
Support Polygon	$\mathcal{P}$	$[-0.07, 0.07]$	(m)
Set of Uncertainties	W	$[-0.05, 0.05]$	(m)
Tracking Sample Period	τ	0.005	(s)
Tracking Gain (22)	k	57.6	(-)
Control Safety Margin	KZ	$[-0.051, 0.051]$	(m)
MPC Sampling Period	T	0.1	(s)
MPC Horizon length	N	16	(-)

It is standard practice to include a saturation in the control law (10) with respect to the constraint (3), either explicitly limiting its value [9], [23] or in an optimization problem [24]. This usually increases significantly the region of attraction of the control law [2]. But this leads to a piecewise linear dynamical behavior, which is more demanding to analyze. In this article, we will limit our analysis to the linear behavior where the saturation is not reached, considering only sets of uncertainties $W \subseteq W_{max}$. Our results are therefore conservative, not considering the improvements due to saturation.

V. MPC SCHEME FOR BIPED WALKING

Humanoid robots are supposed to adapt to dynamically changing environments [3], [1], [8], so the reference walking motion introduced in Section III must be computed online and adapted accordingly. This is usually done with some form of MPC in order to make sure that constraints (18) and (19) are satisfied. We usually want this reference motion to follow some desired trajectory for the CoM and CoP, minimizing an objective function

$$\alpha_1\|p_{ref} - p_d\|^2 + \alpha_2\|c_{ref} - c_d\|^2 + \alpha_3\|\dot{c}_{ref} - \dot{c}_d\|^2, \tag{28}$$

where α_1, α_2 and α_3 are predefined weights. To maintain stability and recursive feasibility, we consider a capturability condition as terminal constraint [22], taking into account the safety margin KZ:

$$\xi_{ref}^{+N} \in \mathcal{P} \ominus \mathcal{N} \ominus KZ. \tag{29}$$

While the tracking control law (10) is sampled every $\tau = 5(\text{ms})$, the reference walking motion is classically recomputed only every $T = 100(\text{ms})$ [3]. Each time it is recomputed, the initial reference state x_{ref}^0 for the prediction can be chosen in different ways. Following the analysis in section III, we know that if the initial tracking error δ^0 is in the set Z,

$$\delta^0 = x^0 - x_{ref}^0 \in Z, \tag{30}$$

where x^0 is the current state of the robot, then the safety margins in (18) and (19) ensure that the control and state constraints (3) and (8) are correctly satisfied.

The reference motion is usually generated with an MPC scheme independently from the actual robot motion [11]. In that case, since the set Z is robust invariant, condition (30) is recursively satisfied and robust recursive feasibility is ensured for all $v \in W$ (see prop.2 in [16]). We are going to see, however, that this choice is too restrictive and can fail tracking desired trajectories even when there are no uncertainties. We will refer to this approach as Open Loop MPC (OL–MPC).

There are situations, anyway, where it is necessary to generate a reference motion taking into account the actual state of the robot [13], such as when there is a strong perturbation. A classical approach is to generate the reference motion starting directly from the actual state of the robot:

$$x_{ref}^0 = x^0. \tag{31}$$

That clearly satisfy the condition (30), but we are going to see that this choice is actually still too restrictive, as desired trajectory tracking can fail even when there are no uncertainties. And when there are uncertainties, recursive feasibility [15] is lost since the initial state (31) may not satisfy constraints (18) or (29), and the controller can diverge. We will refer to this approach as Closed Loop MPC (CL–MPC).

In an approach discussed in [18], the initial state of the reference motion x_{ref}^0 is optimally chosen to take into account the actual state of the robot satisfying (18), (29) and (30) as constraints. Robust recursive feasibility for all $v \in W$ is ensured in this case (see prop.3 in [18]). We will refer to this approach as Robust Closed Loop MPC (RCL–MPC).

VI. SIMULATIONS

We are going to consider now three different scenarios, all involving exclusively lateral motion of the robot, with potential uncertainty v up to ± 0.05 (m), which is significant with respect to the 0.14 (m) wide feet of the robot, in order to illustrate as clearly as possible the differences in behavior obtained with these different MPC schemes. Simulation parameters are provided in Table I.

In the first scenario, the robot is standing still on one foot, without any uncertainty, starting from desired positions of the CoP and CP on the boundary of the control safety margin: $p_d = \xi_d = 0.019$(m). During the simulation, these desired positions are switched to the middle of the foot: $p_d = \xi_d = 0$(m). In this case, in order to move the CP to the middle of the foot, the CoP would have to enter the control safety margin. With both the OL–MPC and the CL–MPC schemes, the reference motion is prohibited to do so. As a result, it can't follow the desired position, and stays stuck instead in its initial position, as shown in Fig. 2. With RCL–MPC, the reference motion is free to start from a different position, slightly away from the boundary of the control safety margin. As a result, the reference motion is able to follow the desired position without problem, as shown in Fig. 3.
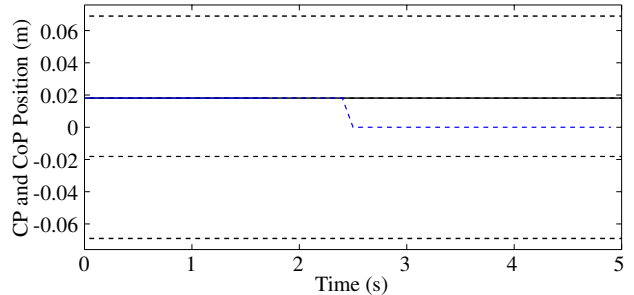

Fig. 2: With both the CL–MPC and the OL–MPC schemes, the CP (black line) and the CoP (blue line) are stuck on the boundary of the control safety margin and can't follow the desired position (blue dashed line) to the middle of the foot.
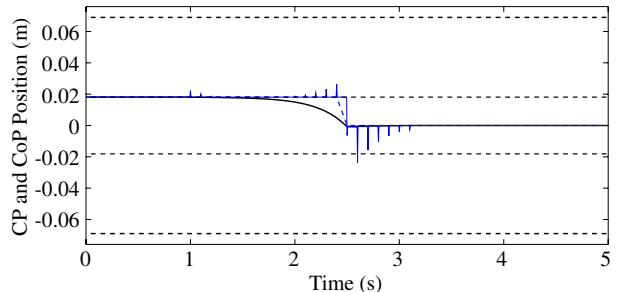

Fig. 3: With the RCL–MPC scheme, the CP (black line) and the CoP (blue line) can follow the desired position (blue dashed line) without problem.

In the second scenario, the robot is again standing still on one foot, starting from desired positions of the CoP and CP on the boundary of the control safety margin: $p_d = \xi_d = 0.019$(m). This time, however, these desired positions are kept constant, and uncertainties are applied instead to the system. After some initial time without uncertainties, a constant, maximal uncertainty is applied for some time, before changing randomly every $T = 0.1$(s), and then every $\tau = 0.005$(s). As soon as the first uncertainty appears, the state of the system is immediately driven slightly inside the control safety margin. At that moment, the terminal constraint in the CL–MPC scheme becomes instantaneously infeasible. If we relax this terminal constraint (as done for example in [3] for a different reason), the state of the system quickly diverges, as shown in Fig. 4. On the contrary, with both the OL–MPC and the RCL–MPC schemes, the reference motion is kept stationary, and safely tracked by the control law (10) as shown in Fig. 5. A zoom on the behavior of the CP is also provided, to show that it stays within bounds

$$\xi - \xi_{ref} \in \begin{bmatrix} 1 & \omega^{-1} \end{bmatrix} Z \tag{32}$$

as expected.

The third scenario involves walking nine consecutive steps before stopping on one foot, and summarizes all the behaviors demonstrated earlier. After a couple steps are realized in place, the desired position is shifted 0.5 (m) on the side.
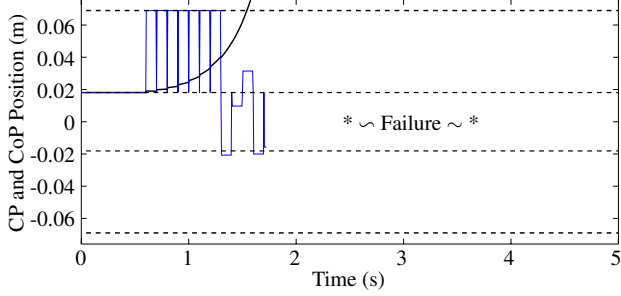
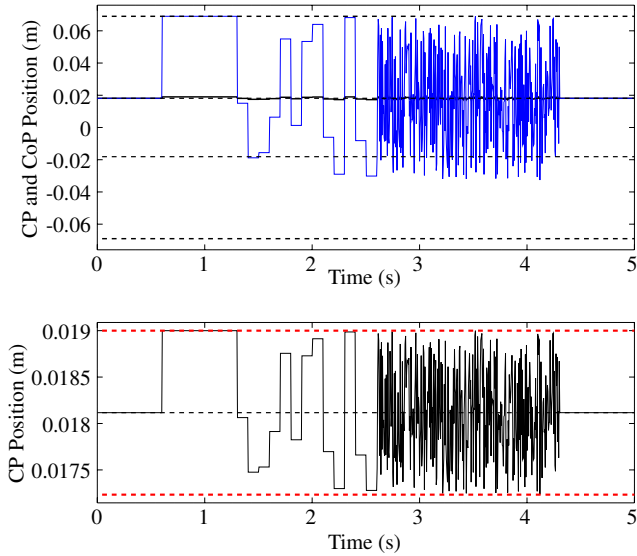Fig. 4: With the CL–MPC scheme, the CP (black line) quickly diverges as soon as the first uncertainty appears.

Fig. 5: With both the OL–MPC and the RCL–MPC schemes, the reference motion is kept stationary, and safely tracked by the control law (10) (top). A zoom on the behavior of the CP (bottom) shows that it stays within bounds $\xi - \xi_{ref} \in [1 \quad \omega^{-1}]Z$ as expected.

After four more steps are realized, a constant uncertainty is applied.

With the OL–MPC scheme (top of Fig. 6), we can observe a significant tracking error (actually an overshoot) with respect to the desired position on the fifth step. With the CL–MPC scheme (middle of Fig. 6), the overshoot is still present, and divergence shortly after the uncertainty is introduced. With the RCL–MPC scheme, both the overshoot and the divergence are avoided (bottom of Fig. 6).

The RCL–MPC scheme clearly shows a much improved behavior over both the CL–MPC and the OL–MPC schemes.

VII. CONCLUSIONS

We analyzed the tracking error dynamics in biped robots affected by bounded uncertainties, and we found the smallest bound possible for the tracking error (the mRPI set) (15). In order to ensure the robot motion recursively satisfies constraints (3) and (8), the reference motion has to be generated
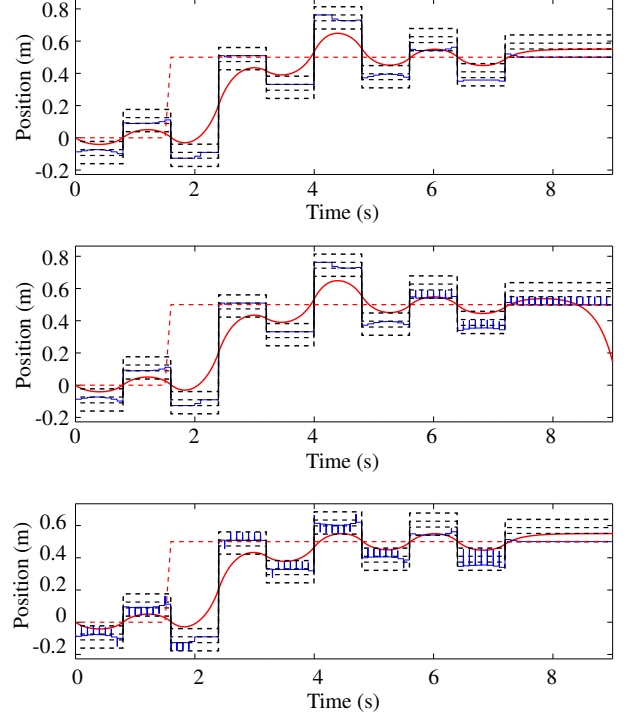

Fig. 6: With the OL–MPC scheme (top), we can observe a significant overshoot with respect to the desired position on the fifth step. With the CL–MPC scheme (middle), the overshoot is still present and divergence shortly after the uncertainty is introduced. With the RCL–MPC scheme, both the overshoot and the divergence are avoided (bottom).

satisfying tighter constraints that consider the tracking error, as shown in (18), (19) and (29).

In [4] and [21] it is proposed a robust approach to handle bounded model errors. The reference motion is optimized to satisfy the system constraints for any possible value of error within these bounds. The present work extends this approach to consider also a tracking error produced by bounded uncertainties.

We proposed a tracking control gain (22) to reduce safety margin size and handle bigger uncertainties. Using parameters corresponding to HRP2 (Table I), and assuming a source of uncertainties in the form of an external horizontal force f, applied on the CoM, the maximal uncertainty supported corresponds to $f_{max} = 48(\text{N})$.

OL–MPC scheme is normally used in biped robots [10] since the correct walking dynamics is ensured for the considered set of uncertainties. In the case of the closed-loop MPC schemes, if the reference motion is generated at each sample time directly from the actual state of the robot (CL–MPC), the controller may fail. This problem is fixed generating the reference motion to satisfy (18), (29) and (30) as constraints (RCL–MPC).

APPENDIX

Let us consider a tracking control gain of the form $K = k \begin{bmatrix} 1 & \lambda \end{bmatrix}$ and look for values k and λ such that

$$K(A + BK) = KA + KBK =$$
$$k \begin{bmatrix} \cosh(\omega\tau) + \lambda\omega\sinh(\omega\tau) & \omega^{-1}\sinh(\omega\tau) + \lambda\cosh(\omega\tau) \end{bmatrix}$$
$$+ k^2(1 - \cosh(\omega\tau) - \lambda\omega\sinh(\omega\tau)) \begin{bmatrix} 1 & \lambda \end{bmatrix} = 0.$$

Multiplying on the right by $\begin{bmatrix} \lambda \\ -1 \end{bmatrix}$ we obtain that $\lambda^2\omega^2 = 1$, the positive solution is $\lambda = \omega^{-1}$. Replacing this value in the previous expression gives $k = \dfrac{e^{\omega\tau}}{e^{\omega\tau} - 1}$.

ACKNOWLEDGMENT

REFERENCES

[1] D. J. Agravante, A. Sherikov, P.-B. Wieber, A. Cherubini, and A. Kheddar. Walking pattern generators designed for physical collaboration. In *Robotics and Automation (ICRA), 2016 IEEE International Conference on*, pages 1573–1578. IEEE, 2016.

[2] C. Atkeson. *Notes on Saturation and Region of Attraction*. Personal notes, 2017.

[3] N. Bohórquez, A. Sherikov, D. Dimitrov, and P.-B. Wieber. Safe navigation strategies for a biped robot walking in a crowd. In *Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on*, pages 379–386. IEEE, 2016.

[4] C. Brasseur, A. Sherikov, C. Collette, D. Dimitrov, and P.-B. Wieber. A robust linear mpc approach to online generation of 3d biped walking motion. In *Humanoid Robots (Humanoids), 2015 IEEE-RAS 15th International Conference on*, pages 595–601. IEEE, 2015.

[5] Y. Choi, D. Kim, and B.-J. You. On the walking control for humanoid robot based on the kinematic resolution of com jacobian with embedded motion. In *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*, pages 2655–2660. IEEE, 2006.

[6] H. Dai and R. Tedrake. Planning robust walking motion on uneven terrain via convex optimization. In *Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on*, pages 579–586. IEEE, 2016.

[7] A. Del Prete and N. Mansard. Robustness to joint-torque-tracking errors in task-space inverse dynamics. *IEEE Transactions on Robotics*, 32(5):1091–1105, 2016.

[8] C. Dune, A. Herdt, E. Marchand, O. Stasse, P. Wieber, and E. Yoshida. Vision based control for humanoid robots. *ViCoMoR*, page 19.

[9] J. Englsberger, C. Ott, M. A. Roa, A. Albu-Schäffer, and G. Hirzinger. Bipedal walking control based on capture point dynamics. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 4420–4427. IEEE, 2011.

[10] S. Feng. Online hierarchical optimization for humanoid control. 2016.

[11] S. Feng, E. Whitman, X. Xinjilefu, and C. G. Atkeson. Optimization based full body control for the atlas robot. In *Humanoid Robots (Humanoids), 2014 14th IEEE-RAS International Conference on*, pages 120–127. IEEE, 2014.

[12] S. Feng, E. Whitman, X. Xinjilefu, and C. G. Atkeson. Optimization-based full body control for the darpa robotics challenge. *Journal of Field Robotics*, 32(2):293–312, 2015.

[13] S. Feng, X. Xinjilefu, C. G. Atkeson, and J. Kim. Robust dynamic walking using online foot step optimization. In *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, pages 5373–5378. IEEE, 2016.

[14] A. Herdt, H. Diedam, P.-B. Wieber, D. Dimitrov, K. Mombaur, and M. Diehl. Online walking motion generation with automatic footstep placement. *Advanced Robotics*, 24(5-6):719–737, 2010.

[15] E. C. Kerrigan. *Robust constraint satisfaction: Invariant sets and predictive control*. PhD thesis, Citeseer, 2001.

[16] W. Langson, I. Chryssochoos, S. Raković, and D. Q. Mayne. Robust model predictive control using tubes. *Automatica*, 40(1):125–133, 2004.

[17] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. Scokaert. Constrained model predictive control: Stability and optimality. *Automatica*, 36(6):789–814, 2000.

[18] D. Q. Mayne, M. M. Seron, and S. Raković. Robust model predictive control of constrained linear systems with bounded disturbances. *Automatica*, 41(2):219–224, 2005.

[19] K. Ogata. *Discrete-time control systems*, volume 8. Prentice-Hall Englewood Cliffs, NJ, 1995.

[20] S. V. Rakovic, E. C. Kerrigan, K. I. Kouramas, and D. Q. Mayne. Invariant approximations of the minimal robust positively invariant set. *IEEE Transactions on Automatic Control*, 50(3):406–410, 2005.

[21] D. Serra, C. Brasseur, A. Sherikov, D. Dimitrov, and P.-B. Wieber. A newton method with always feasible iterates for nonlinear model predictive control of walking in a multi-contact situation. In *Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on*, pages 932–937. IEEE, 2016.

[22] A. Sherikov. *Balance preservation and task prioritization in whole body motion control of humanoid robots*. PhD thesis, Grenoble Alpes, 2016.

[23] T. Sugihara. Standing stabilizability and stepping maneuver in planar bipedalism based on the best com-zmp regulator. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 1966–1971. IEEE, 2009.

[24] P.-B. Wieber. Trajectory free linear model predictive control for stable walking in the presence of strong perturbations. In *Humanoid Robots, 2006 6th IEEE-RAS International Conference on*, pages 137–142. IEEE, 2006.

[25] P.-B. Wieber, R. Tedrake, and S. Kuindersma. Modeling and control of legged robots. In *Springer Handbook of Robotics*, pages 1203–1234. Springer, 2016.