HAL
open science

Online Walking Gait Generation with Adaptive Foot Positioning through Linear Model Predictive Control

Holger Diedam, Dimitar Dimitrov, Pierre-Brice Wieber, Katja Mombaur, Moritz Diehl

To cite this version:

Holger Diedam, Dimitar Dimitrov, Pierre-Brice Wieber, Katja Mombaur, Moritz Diehl. Online Walking Gait Generation with Adaptive Foot Positioning through Linear Model Predictive Control. IROS 2008 - IEEE-RSJ International Conference on Intelligent Robots & Systems, Sep 2008, Nice, France. pp.1121-1126, 10.1109/IROS.2008.4651055 . inria-00390566

HAL Id: inria-00390566

https://inria.hal.science/inria-00390566

Submitted on 2 Jun 2009

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Online Walking Gait Generation with Adaptive Foot Positioning through Linear Model Predictive Control

Holger Diedam Dimitar Dimitrov Pierre-Brice Wieber Katja Mombaur Moritz Diehl

***Abstract*— Building on previous propositions to generate walking gaits online through the use of Linear Model Predictive Control, the goal of this paper is to show that it is possible to allow on top of that a continuous adaptation of the positions of the foot steps, allowing the generation of stable walking gaits even in the presence of strong perturbations, and that this additional adaptation requires only a minimal modification of the previous schemes, especially maintaining the same Linear Model Predictive form. Simulation results are proposed then on the HRP-2 humanoid robot, showing a significant improvement over the previous schemes.**

I. INTRODUCTION

The difficulty in generating a stable walking motion mostly lies in the fact that moving one's Center of Mass (CoM) entirely relies on the contact forces between the feet and the ground, with the constraint that feet can only push on the ground [13], [14]. This restricts the motions that a walking system can realize, strongly limiting its capacity to follow a predefined motion in the presence of perturbations [17]. There is a strong interest therefore in being able to generate walking motions online, continuously adapting them to the current dynamics of the system.

A promising approach making use of a Linear Quadratic Regulator (LQR) has been proposed in [5]. Based on a linear approximation of the dynamics of the system, this approach tries to keep the contact forces in the middle of the feasible set, stabilizing the motion of the CoM of the system by minimizing its jerk over a finite prediction horizon. This LQR based approach allows generating stable walking motions online, with the possibility to continuously take into account the current state of the system [6], [8]. But its intrinsically linear conception not considering explicitly the constraints on the contact forces seriously limits its capacity to deal with strong perturbations.

In order to overcome this limitation, it has been proposed to introduce these constraints explicitly into the regulator, turning the LQR scheme into a more general Linear Model Predictive Control (LMPC) scheme, what lead to significant improvements in its capacity to deal with strong perturbations [15]. But both of these propositions were designed to work with foot step positions decided and fixed beforehand by a foot step planner, what unnecessarily limits the robot's capacity to deal with perturbations since being able to adapt step positions online can contribute significantly to dealing with perturbations.

Hopefully, this MPC scheme appears to be very general in its capacity to generate stable walking motions, its only mandatory feature being the regulation of the amplitude of derivatives of the motion of the CoM over a prediction horizon [16]. Any control variable contributing to this goal can be instrumental in generating ever more stable walking motions. Following this analysis, we propose here to introduce new control variables corresponding to the positions of the foot steps, allowing their adaptation with only a minimal addition to this already well known LMPC scheme.

Note that the idea of adapting online the foot steps of a walking robot has already been investigated thoroughly in the specific case of stopping motions [7], [9], but with solutions which are not obvious to apply to more general walking gaits.

The paper is organized as follows: Section II will present the LMPC scheme previously introduced in [5], [15], while Section III will present how it can be simply modified in order to incorporate a continuous adaptation of the feet positions. Important details about how the constraints on the Center of Pressure (CoP) need to be addressed will be given in Section IV before giving simulation results and concluding.

II. THE PREDICTIVE CONTROL SCHEME

Following the ideas introduced in [5], [15], the predictive control presented in this section will focus on the motion of the CoM of the walking robot. Even though Nonlinear MPC schemes are getting more and more accessible to fast systems requiring short computation times such as walking robots, thanks to state of the art mathematical methods [1], Linear MPC schemes still allow shorter computation times and therefore faster control loops, what is highly desirable. In order to obtain a Linear MPC scheme here, we will begin by assuming that the robot walks on a constant horizontal plane, and that the motion of its CoM is also constrained to a horizontal plane at a distance h above the ground, so that its position in space can be defined using only two variables (x, y).

We will consider trajectories of the CoM which have piecewise constant jerks $\dddot{x}$ and $\dddot{y}$ over time intervals of constant length T so that we can compute the corresponding dynamics at discrete times t_k (here only expressed for the coordinate x of the CoM, but the same applies to the

H. Diedam and K. Mombaur are with Interdisciplinary Center for Scientific Computing, University of Heidelberg, Im Neuenheimer Feld 368, D-69120 Heidelberg, Germany hdiedam@ix.urz.uni-heidelberg.de, kmombaur@uni-hd.de

D. Dimitrov and P.-B. Wieber are with INRIA Grenoble - Rhône-Alpes, Inovallée, 38334 Saint Ismier Cedex, France Pierre-Brice.Wieber@inrialpes.fr

M. Diehl is with ESAT, K.U. Leuven, Kasteelpark Arenberg 10, 3001 Leuven-Heverlee, Belgium Moritz.Diehl@esat.kuleuven.be

coordinate y):

$$\hat{x}_{k+1} = A\,\hat{x}_k + B\,\dddot{x}(t_k) \tag{1}$$

with

$$\hat{x}_k = \begin{pmatrix} x(t_k) \\ \dot{x}(t_k) \\ \ddot{x}(t_k) \end{pmatrix} \tag{2}$$

and

$$A = \begin{pmatrix} 1 & T & T^2/2 \\ 0 & 1 & T \\ 0 & 0 & 1 \end{pmatrix},\ B = \begin{pmatrix} T^3/6 \\ T^2/2 \\ T \end{pmatrix}. \tag{3}$$

We will consider an approximation of the position (z^x, z^y) of the CoP on the ground (also know as the Zero Moment Point, ZMP [13]) corresponding to this motion by neglecting the inertial effects due to rotations of the different parts of the robot:

$$z_k^x = \begin{pmatrix} 1 & 0 & -h/g \end{pmatrix} \hat{x}_k \tag{4}$$

with h the constant height of the CoM above the ground and g the norm of the gravity force.

Using the dynamics (1) recursively, we can derive relationships between the jerk of the CoM, its position and velocity and the position of the CoP over longer time intervals, of length NT:

$$X_{k+1} = \begin{pmatrix} x_{k+1} \\ \vdots \\ x_{k+N} \end{pmatrix} = P_{ps}\hat{x}_k + P_{pu}\dddot{X}_k, \tag{5}$$

$$\dot{X}_{k+1} = \begin{pmatrix} \dot{x}_{k+1} \\ \vdots \\ \dot{x}_{k+N} \end{pmatrix} = P_{vs}\hat{x}_k + P_{vu}\dddot{X}_k \tag{6}$$

and

$$Z_{k+1}^x = \begin{pmatrix} z_{k+1}^x \\ \vdots \\ z_{k+N}^x \end{pmatrix} = P_{zs}\hat{x}_k + P_{zu}\dddot{X}_k \tag{7}$$

with

$$\dddot{X}_k = \begin{pmatrix} \dddot{x}_k \\ \vdots \\ \dddot{x}_{k+N-1} \end{pmatrix}. \tag{8}$$

The matrices P_{ps}, P_{vs}, $P_{zs} \in \mathbb{R}^{N\times 3}$ and P_{pu}, P_{vu}, $P_{zu} \in \mathbb{R}^{N\times N}$ introduced here follow directly from the recursive application of the dynamics (1).

The MPC scheme introduced in [5], [15] amounts then to computing the trajectory of the CoM which minimizes the following function over a prediction horizon of length NT:

$$\min_{\dddot{X}_k, \dddot{Y}_k} \frac{1}{2}\left(\left\|\dddot{X}_k\right\|^2 + \left\|\dddot{Y}_k\right\|^2\right) + \frac{\alpha}{2}\left(\left\|\dot{X}_{k+1}\right\|^2 + \left\|\dot{Y}_{k+1}\right\|^2\right) + \frac{\beta}{2}\left(\left\|X_{k+1} - X_{k+1}^{ref}\right\|^2 + \left\|Y_{k+1} - Y_{k+1}^{ref}\right\|^2\right) \tag{9}$$

while considering constraints on the CoP that will be discussed at more length in section IV.

The parts of this objective function implying the position and velocity of the CoM introduce an a priori on its desired motion which is not mandatory. They drive the CoM of the robot with a bit of damping towards a reference position (X_k^{ref}, Y_k^{ref}) which is decided to lie in the middle of the support polygon, what can result in trajectories slightly more robust with respect to perturbations. Solely minimizing higher derivatives can be shown however to be sufficient for generating stable walking motions [16]. The choice of the constants α and β appears therefore to be largely open and depend on the desired a priori. The computations in this paper have been performed with $\alpha = 200\,s^{-4}$ and $\beta = 1000\,s^{-6}$ for $N = 15$ time intervals of length $T = 0.1\,s$.

This minimization problem can be expressed then as a canonical Quadratic Program (QP)

$$\min_{u_k} \frac{1}{2}u_k^T Q u_k + p_k^T u_k \tag{10}$$

with

$$u_k = \begin{pmatrix} \dddot{X}_k \\ \dddot{Y}_k \end{pmatrix}, \tag{11}$$

$$Q = \begin{pmatrix} Q' & 0 \\ 0 & Q' \end{pmatrix}, \tag{12}$$

$$Q' = I + \alpha P_{vu}^T P_{vu} + \beta P_{pu}^T P_{pu} \tag{13}$$

and

$$p_k = \begin{pmatrix} \alpha\hat{x}_k^T P_{vs}^T P_{vu} + \beta\hat{x}_k^T P_{ps}^T P_{pu} - \beta(X_{k+1}^{ref})^T P_{pu} \\ \alpha\hat{y}_k^T P_{vs}^T P_{vu} + \beta\hat{y}_k^T P_{ps}^T P_{pu} - \beta(Y_{k+1}^{ref})^T P_{pu} \end{pmatrix}. \tag{14}$$

Observe that the matrix Q here is constant with time, so it can be prefactorized to minimize online computation time.

III. ADAPTING STEP POSITIONS

The MPC scheme presented in the previous section has been exclusively used so far to generate trajectories of the CoM with step positions decided beforehand by a step planner [5], [6], [8], [12], [15]. This MPC scheme can be shown however to be far more general in its capacity to generate stable walking motions, its only mandatory feature being the regulation of the amplitude of derivatives of the motion of the CoM over a prediction horizon [16]. Any control variable contributing to this goal can be instrumental in generating ever more stable walking motions. Following this analysis, we propose here to let the step positions be also decided by the MPC scheme by simply adding new variables (X_k^f, Y_k^f) corresponding to the positions of the m foot steps occurring over the prediction horizon.

Letting the step positions free, we need to introduce a new way to steer the robot towards a desired direction, either in the objective function (9) or through constraints. We propose here to use the same step planner as earlier to generate reference steps. These reference steps are handled then to the MPC scheme which will try to execute them, keeping however a complete freedom in the final choice of the step positions according to the robot stability and mechanical limits. This is easily done with an additional cost to the function (9) driving the choice of the step positions towards

pre-generated reference step positions $(X_k^{f\text{-}ref}, Y_k^{f\text{-}ref})$, resulting in the QP

$$\min_{u_k} \frac{1}{2}\left(\left\|\dddot{X}_k\right\|^2 + \left\|\dddot{Y}_k\right\|^2\right) + \frac{\alpha}{2}\left(\left\|\dot{X}_{k+1}\right\|^2 + \left\|\dot{Y}_{k+1}\right\|^2\right) + \frac{\beta}{2}\left(\left\|X_{k+1} - X_{k+1}^{ref}\right\|^2 + \left\|Y_{k+1} - Y_{k+1}^{ref}\right\|^2\right) + \frac{\gamma}{2}\left(\left\|X_k^f - X_k^{f\text{-}ref}\right\|^2 + \left\|Y_k^f - Y_k^{f\text{-}ref}\right\|^2\right) \quad (15)$$

with

$$u_k = \begin{pmatrix} \dddot{X}_k \\ X_k^f \\ \dddot{Y}_k \\ Y_k^f \end{pmatrix}. \quad (16)$$

Since the reference position (X_k^{ref}, Y_k^{ref}) of the CoM that appears in this QP has been decided to lie in the middle of the support polygon, it naturally depends on the actual step positions (X_k^f, Y_k^f). If we consider that these step positions directly indicate the center of each foot's support polygon, and if we consider that none of the sampling times t_k falls strictly inside a double support phase, what will be discussed in more depth in the next section, we immediately obtain that

$$X_{k+1}^{ref} = U_{k+1} X_k^f \quad (17)$$

with

$$U_{k+1} := \begin{pmatrix} 1 & 0 & \\ \vdots & \vdots & \\ 1 & 0 & \\ 0 & 1 & \\ \vdots & \vdots & \\ 0 & 1 & \ddots \\ 0 & 0 & \\ \vdots & \vdots & \\ 0 & 0 & \end{pmatrix}. \quad (18)$$

The ones in this matrix $U_{k+1} \in \mathbb{R}^{N \times m}$ simply indicate which sampling times t_k fall into which step, where sampling times correspond to rows and steps to column.

We can express then this new QP in the same canonical form (10), but this time with a varying quadratic term because of the varying matrix U_{k+1}:

$$Q_k = \begin{pmatrix} Q'_k & 0 \\ 0 & Q'_k \end{pmatrix}, \quad (19)$$

$$Q'_k = \begin{pmatrix} I + \alpha P_{vu}^T P_{vu} + \beta P_{pu}^T P_{pu} & -\beta P_{pu}^T U_{k+1} \\ -\beta P_{pu}^T U_{k+1} & \beta U_{k+1}^T U_{k+1} + \gamma I \end{pmatrix} \quad (20)$$

and

$$p_k = \begin{pmatrix} \alpha \hat{x}_k^T P_{vs}^T P_{vu} + \beta \hat{x}_k^T P_{ps}^T P_{pu} \\ -\beta \hat{x}_k^T P_{ps}^T U_{k+1} - \gamma X_k^{f\text{-}ref} \\ \alpha \hat{y}_k^T P_{vs}^T P_{vu} + \beta \hat{y}_k^T P_{ps}^T P_{pu} \\ -\beta \hat{y}_k^T P_{ps}^T U_{k+1} - \gamma Y_k^{f\text{-}ref} \end{pmatrix}. \quad (21)$$

Hopefully, the matrix U_{k+1} varies cyclically with time so prefactorizing a whole cycle of matrices Q_k would still be possible to minimize online computation time.

An alternative is to relate the reference position of the CoM not to the actual center of the support polygon but to the one of the reference foot steps, replacing (17) with

$$X_{k+1}^{ref} = U_{k+1} X_k^{f\text{-}ref}. \quad (22)$$

In this case, the definition (19)-(21) of the QP turns into

$$Q = \begin{pmatrix} Q' & 0 \\ 0 & Q' \end{pmatrix}, \quad (23)$$

$$Q' = \begin{pmatrix} I + \alpha P_{vu}^T P_{vu} + \beta P_{pu}^T P_{pu} & 0 \\ 0 & \gamma I \end{pmatrix} \quad (24)$$

and

$$p_k = \begin{pmatrix} \alpha \hat{x}_k^T P_{vs}^T P_{vu} + \beta \hat{x}_k^T P_{ps}^T P_{pu} - \beta (X_k^{f\text{-}ref})^T U_{k+1}^T P_{pu} \\ -\gamma X_k^{f\text{-}ref} \\ \alpha \hat{y}_k^T P_{vs}^T P_{vu} + \beta \hat{y}_k^T P_{ps}^T P_{pu} - \beta (Y_{k+1}^{f\text{-}ref})^T U_{k+1}^T P_{pu} \\ -\gamma Y_k^{f\text{-}ref} \end{pmatrix}, \quad (25)$$

bringing back a constant matrix Q. We can observe also that steering the CoM towards the center of the reference support polygon instead of the center of the actual support polygon goes along the initial idea of using the reference foot steps for driving the motion of the robot. Anyway, these terms are of secondary importance for the stability of the generated gait as discussed in section II, so the choice between these two alternatives should be considered also of secondary importance. The results in this paper have been obtained with the second alternative.

IV. CONSTRAINTS ON THE CENTER OF PRESSURE

The definition of the MPC scheme in section II wouldn't be complete without specifying the constraints restricting the position of the CoP on the ground. Since the feet of the robot can only push on the ground, this CoP can only lie within the support polygon, the convex hull of the contact points between the feet and the ground [13]. Any trajectory not satisfying this constraint wouldn't be realizable.

During single support, considering a foot with a polygonal shape on the ground at a position (x^f, y^f) with an orientation θ, this constraint can be expressed as a set of linear constraints

$$\begin{pmatrix} d_x(\theta) & d_y(\theta) \end{pmatrix} \begin{pmatrix} z^x - x^f \\ z^y - y^f \end{pmatrix} \leq b(\theta) \quad (26)$$

on the position of the CoP. We can observe that these inequalities are linear with respect to the position of the foot on the ground but nonlinear with respect to its orientation. In the case of double support, these inequalities won't be linear anymore with respect to the feet position but quadratic because of the cross-products hidden in the computation of

the vectors d_x and d_y. Such nonlinearities are best avoided in order to keep the form of a QP with linear constraints which is very advantageous from a computational point of view. For this reason, we will consider here the orientations of the feet decided in advance and we will discuss now in more depth the question of double support.

It appears that satisfying the constraint on the position of the CoP only at discrete instants is largely enough for generating realizable motions, under the mandatory condition that it is satisfied at all transition times during single and double support phases (and introducing a safety margin as presented later in section VI). An important observation is that at these transition times, the constraints of both single and double support apply, but those of single support are the most restrictive and are therefore sufficient. We choose here to satisfy the constraint on the position of the CoP at the sampling times t_k, with a period T in between chosen to be strictly equal to the length of the double support phases ($0.1\,s$ here, with single support periods of $0.7\,s$) so that no sampling time falls strictly inside them. This way, we end up having to consider only the single support constraint (26).

Expressing now this constraint at the instants t_k over the whole prediction horizon leads to

$$D_{k+1}\begin{pmatrix} Z^x_{k+1} - U_{k+1}X^f_k \\ Z^y_{k+1} - U_{k+1}Y^f_k \end{pmatrix} \leq b_{k+1}, \tag{27}$$

or with respect to the vector u_k defined in (16),

$$D_{k+1}\begin{pmatrix} P_{zu} & -U_{k+1} & 0 & 0 \\ 0 & 0 & P_{zu} & -U_{k+1} \end{pmatrix} u_k \leq b_{k+1} - \begin{pmatrix} P_{zs}\hat{x}_k \\ P_{zs}\hat{y}_k \end{pmatrix}, \tag{28}$$

with D_{k+1} of the following simple double diagonal form,

$$D_{k+1} = \begin{pmatrix} d_x(\theta_1) & 0 & \dots & 0 & d_y(\theta_1) & \dots \\ 0 & d_x(\theta_1) & \ddots & \vdots & 0 & \ddots \\ \vdots & \ddots & \ddots & 0 & \vdots & \ddots \\ 0 & \dots & 0 & d_x(\theta_m) & 0 & \dots \end{pmatrix}. \tag{29}$$

Due to the special structure of these matrices, no matrix multiplications are required for assembling the final inequality constraint (28), what can be done therefore very quickly. On top of that, their evolution in time is also highly structured, so this assembling need not even be realized at each time.

V. CONSTRAINTS ON THE FOOT POSITION

Allowing modifications of the foot positions by this MPC scheme, we need to be sure that these modifications won't lead to unrealizable motions because of the geometric and kinematic limitations of the robot: maximum leg length, joint limits, self-collision avoidance, maximum joint speed and other similar limitations need to be taken care of. The only thing we need to do for this is to derive simple approximations of all these limitations that can be expressed in the form of linear constraints on the vector u_k defined in (16).

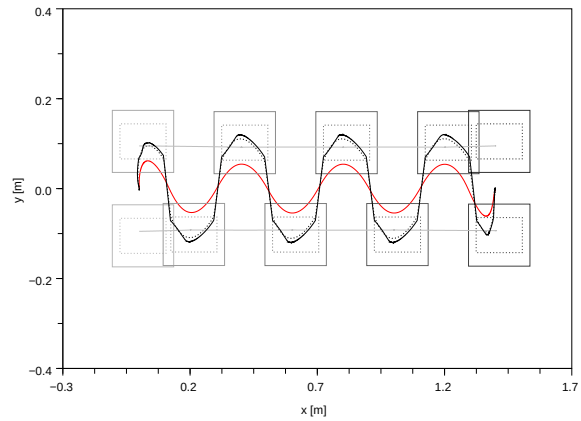

Fig. 1. Motion of the CoM (in red) and CoP (in black) and foot steps generated by our MPC scheme when the system is not submitted to any perturbation. The dotted plots correspond to the CoP computed with the approximate model together with the safety margins $3\,cm$ inside the foot prints used in the QP computations. The continuous plots correspond to the real CoP of the robot together with the real foot prints. Note an evolution of the CoP from heels to toes very similar to what can be observed in humans.

We can derive for example simple linear bounds on the positions of the feet one with respect to the other with minimum and maximum values preventing collision on one side and over-stretching of the legs on the other side. Concerning maximum joint speed, we have found that a very simple bound on the position of the next foot step depending on the current position of the foot in the air and a simple Cartesian maximum speed gives good results:

$$C\begin{pmatrix} \left(X^f_k\right)_1 - x^f(t) \\ \left(Y^f_k\right)_1 - y^f(t) \end{pmatrix} \leq (t_{touchdown} - t)v_{max} \tag{30}$$

with $\left(X^f_k, Y^f_k\right)_1$ the position of the next foot step, $\left(x^f(t), y^f(t)\right)$ the current position of the foot in the air, v_{max} a vector of approximate maximum Cartesian speed in the directions indicated by the matrix C and $t_{touchdown}$ the time when the foot in the air is planned to touch the ground. The computations in this paper have been obtained with maximum speeds of $0.43\,m.s^{-1}$ in the forward direction and $0.14\,m.s^{-1}$ in the lateral direction.

VI. SIMULATION RESULTS

Fig. 1 shows a walking gait generated by the MPC scheme presented in the previous sections when the system is not submitted to any perturbation. The foot steps appear to be only very slightly modified with respect to their reference positions, around $2\,mm$ towards the inside. We can observe an evolution of the CoP from heels to toes very similar to what can be observed in humans. It doesn't reach the toes though since we are not considering toe joint rotation phases here, but this is solely due to our choice of parameters and this gait generation scheme should be able to deal seamlessly with such phases of motion. This motion, as well as the

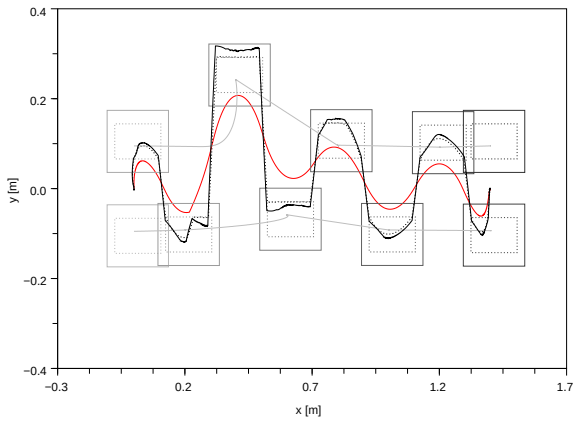

Fig. 2. Same plot as in Fig. 1 but with a system submitted to a perturbation in the lateral direction in the middle of the second single support phase. The motion of the feet in the air is also shown in grey. We can observe a very large recovery step aside and a convergence back to the reference motion in 4 steps, with a CoP continuously staying within the boundaries of the support polygon, as required. We can observe that the approximate CoP continuously stays on the edge of the safety margin for at least 3 steps after the perturbation occurs, an indication that the amplitude of this perturbation is not far from the limits of what this scheme can compensate.

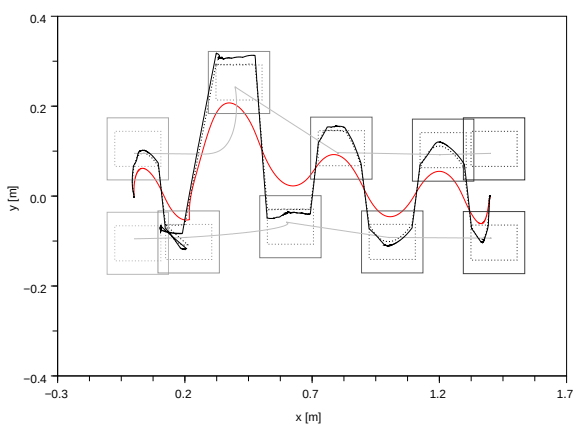

Fig. 3. Same plot as before but with a system submitted to a perturbation in the lateral and backward direction in the middle of the second single support phase. Once again, the CoP continuously stays within the boundaries of the support polygon as required.

following ones, has been generated and verified in simulation with a complete multi-body dynamical model of the HRP-2 humanoid robot.

Fig. 2 shows the outcome of a perturbation happening in the lateral direction in the middle of the second single support phase, and Fig. 3 the outcome of a similar perturbation but in a diagonal direction, lateral and backwards. We can observe in both cases a recovery motion beginning with one big recovery step followed by a convergence back to the reference motion in 3 or 4 steps. We can observe that the CoP continuously lies within the boundaries of the support polygon, as required. In fact the constraints on the CoP are exactly satisfied only at the instants t_k, and they appear to be slightly violated sometimes between these instants, but we have used a safety margin of $3\,cm$ inside the true boundaries of the support polygon so as to be completely safe with respect to this discretization effect.

The trajectories of the feet in the air can also be observed on these figures, showing how the continuous adaptation of the foot steps gives rise to continuously smooth global motions, with a final modification of the first recovery step as big as $18\,cm$.

These perturbations were computed as the results of a ball falling from $0.5\,m$ high around a pivot and hitting the robot horizontally at the height of its CoM through a plastic impact – with complete transfer of the kinetic energy. A decoupling Task Function control law [18] allows compensating independently the perturbations induced on the CoM and on the different limbs of the robot. These perturbations correspond to a mass of the ball equivalent to up to $13\,\%$ of the mass of the robot. As a comparison, such impacts with weight ratios bigger than $7\,\%$ can hardly be compensated by the MPC schemes without adaptive foot positioning under the same conditions [6], [15]. The improvement is therefore significant. Note however that the adaptive foot positioning scheme proposed here applies so far only to continuous walking motions: it allows compensating perturbations while walking, not in other cases such as when standing still.

Assembling and solving the QPs in all these cases took less than $1\,ms$ in average with state of the art solvers such as QL [11] or qpOASES [4], [3], [10], without any serious optimization of the code and notwithstanding observations such those presented in [2] which can help greatly improve computation time.

VII. Conclusion

We have presented an extension to an LMPC scheme already proposed previously for generating walking gaits online. This extension amounts to adding new variables to the problem representing the future foot step positions. Doing so, the general form of the numerical scheme is kept identical while leading to significantly improved capacities to compensate strong perturbations. Indeed, simulations on the HRP-2 humanoid robot show that this modified scheme allows compensating perturbations twice stronger than earlier versions. Demonstrating these results on the real platform should be the next step now.

VIII. Acknowledgments

The first and third authors were supported by the German-French cooperation program Procope coordinated by the German Academic Exchange Service (DAAD) and the Programme Hubert Curien.

The fourth author has been supported by the Postdoc Program of the foundation Landesstiftung Baden-Württemberg.

The fifth author gratefully acknowledges financial support by the Research Council KUL (Center of

Excellence on Optimization in Engineering (OPTEC) EF/05/006, GOA AMBioRICS, IOF-SCORES4CHEM and PhD/postdoc/fellow grants), the Flemish Government via FWO (PhD/postdoc grants, projects G.0452.04, G.0499.04, G.0211.05, G.0226.06, G.0321.06, G.0302.07, research communities ICCoS, ANMMM, MLDM) and via IWT (PhD Grants, McKnow-E, Eureka-Flite), the EU via ERNSI, as well as the Belgian Federal Science Policy Office: IUAP P6/04 (DYSCO, Dynamical systems, control and optimization, 2007-2011).

REFERENCES

[1] M. Diehl. *Real-Time Optimization for Large Scale Nonlinear Processes*. PhD thesis, Universität Heidelberg, 2001.

[2] D. Dimitrov, J. Ferreau, P.-B. Wieber, and M. Diehl. On the implementation of model predictive control for on-line walking pattern generation. In *Proceedings of the IEEE International Conference on Robotics & Automation*, 2008.

[3] H.J. Ferreau, H.G. Bock, and M. Diehl. An online active set strategy to overcome the limitations of explicit MPC. *International Journal of Robust and Nonlinear Control*, 18(8):816–830, 2008.

[4] H.J. Ferreau, P. Ortner, P. Langthaler, L. del Re, and M. Diehl. Predictive control of a real-world diesel engine using an extended online active set strategy. *Annual Reviews in Control*, 31(2):293–301, 2007.

[5] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa. Biped walking pattern generation by using preview control of zero-moment point. In *Proceedings of the IEEE International Conference on Robotics & Automation*, 2003.

[6] S. Kanzaki, K. Nishiwaki, K. Okada, and M. Inaba. Bracing against impact in a humanoid using disturbance preview control. In *Proceedings of the Annual Conference on Robotics Society of Japan*, 2004.

[7] M. Morisawa, S. Kajita, K. Harada, K. Fujiwara, F. Kanehiro, K. Kaneko, and H. Hirukawa. Emergency stop algorithm for walking humanoid robots. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots & Systems*, 2005.

[8] K. Nishiwaki and S. Kagami. High frequency walking pattern generation based on preview control of zmp. In *Proceedings of the IEEE International Conference on Robotics & Automation*, 2006.

[9] J. Pratt, J. Carff, S. Drakunov, and A. Goswami. Capture point: A step toward humanoid push recovery. In *Proceedings of the International Conference on Humanoid Robotics*, 2006.

[10] http://homes.esat.kuleuven.be/~optec/software/qpoases/.

[11] K. Schittkowski. QL: A fortran code for convex quadratic programming - user's guide, version 2.11. Research Report, Department of Mathematics, University of Bayreuth, 2005.

[12] H. Takeuchi. Real time optimization for robot control using receding horizon control with equal constraint. *Journal of Robotic Systems*, 20(1):3–13, 2003.

[13] P.-B. Wieber. On the stability of walking systems. In *Proceedings of the International Workshop on Humanoid and Human Friendly Robotics*, 2002.

[14] P.-B. Wieber. Holonomy and nonholonomy in the dynamics of articulated motion. In *Proceedings of the Ruperto Carola Symposium on Fast Motion in Biomechanics and Robotics*, 2005.

[15] P.-B. Wieber. Trajectory free linear model predictive control for stable walking in the presence of strong perturbations. In *Proceedings of the International Conference on Humanoid Robotics*, 2006.

[16] P.-B. Wieber. Viability and predictive control for safe locomotion. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots & Systems*, 2008.

[17] P.-B. Wieber and C. Chevallereau. Online adaptation of reference trajectories for the control of walking systems. *Robotics and Autonomous Systems*, 54(7), 2006.

[18] P.B. Wieber. *Modélisation et commande d'un robot marcheur anthropomorphe*. PhD thesis, Mines de Paris, 2000.