

Online Non-linear Centroidal MPC for Humanoid Robots Payload Carrying with Contact-Stable Force Parametrization

Mohamed Elobaid¹, Giulio Romualdi¹, Gabriele Nava¹, Lorenzo Rapetti^{1,3},
 Hosameldin Awadalla Omer Mohamed^{1,2} and Daniele Pucci^{1,3}

Abstract—In this paper we consider the problem of allowing a humanoid robot that is subject to a persistent disturbance, in the form of a payload-carrying task, to follow given planned footsteps. To solve this problem, we combine an online non-linear centroidal Model Predictive Controller - MPC with a contact stable force parametrization. The cost function of the MPC is augmented with terms handling the disturbance and regularizing the parameter. The performance of the resulting controller is validated both in simulations and on the humanoid robot iCub. Finally, the effect of using the parametrization on the computational time of the controller is briefly studied.

I. INTRODUCTION

Of the various humanoid robots locomotion control architectures, a modular model-based design emphasizing the separation between high-level trajectory generation and adjustment block(s) and a whole-body trajectory tracking block is becoming a mainstay [1]. The high-level control layer typically utilizes “template” models to reason about the center of mass and feet trajectories [2], while the whole-body control layer uses the robot full model to track the adapted trajectories (see Figure 1). This paper focuses on designing a high-level trajectory adjustment controller leveraging a template model to allow for humanoid robots locomotion under the action of persistent disturbances.

When using “template” models for high-level objectives, the Linear Inverted Pendulum (and its variations) [3] is typically used in an optimal control formulation [4]. However, while this approach proved very successful attested by the large body of literature on the subject [5]–[7], it limits the engineer both in terms of how dynamic the walking pattern achieved is (specially for position controlled robots) as well as the different tasks possible (e.g. running and jumping). In this context reduced models (of which Orin’s and Goswami’s centroidal momentum dynamics model [8] is perhaps the most relevant to our work) are more suited to different balancing and locomotion tasks.

Moreover, as humanoid robots are increasingly expected to carry out collaborative tasks in a work environment [9], research on *robust locomotion* for humanoid robots is gaining more interest [10]–[12]. The collaborative nature of the work entails studying not only impulsive disturbances effects on the humanoid balancing and locomotion (e.g., a push on the

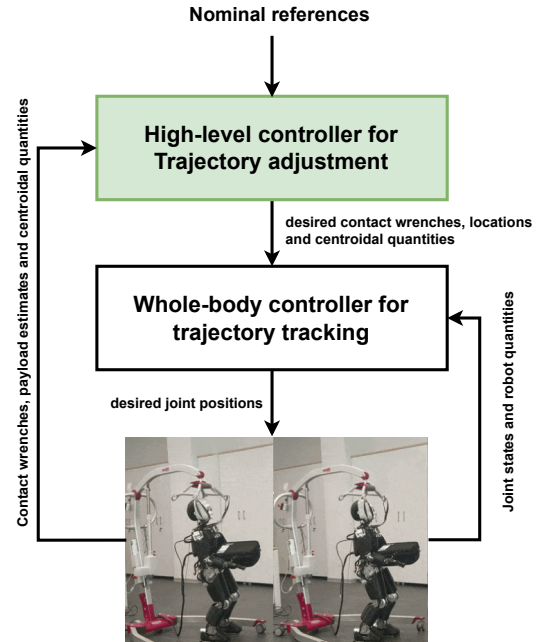


Fig. 1: The controller highlighted in a typical multi-layer bipedal locomotion control architecture.

humanoid robot [13], [14]), but also *persistent* disturbances (e.g., a payload being carried by the robot [15]–[18]).

Several works in the literature deal with persistent disturbances in quadruped (e.g. [19]) and humanoids locomotion (e.g. [20]–[22]). In particular, they can be classified in two broad categories; the first approach deals with the disturbance via robust control machinery. An example of this is the use of robust tube-based MPC as in the work of Villa and Wieber [21].

The second (and more relevant to this work) approach relies on leveraging some knowledge about the persistent disturbance in the controller design. This knowledge may be obtained via the use of disturbance observers and/or estimators as in the works of Smaldone et. al [22], Kaneko et. al [23] and the references therein.

In this direction, this paper contributes towards allowing a humanoid robot to carry a payload while walking, leveraging knowledge of the payload effects on the centroidal dynamics quantities. More in details, building upon [2], we propose an online non-linear MPC *payload-aware* high-level controller. In addition, we simplify the online non-linear optimization problem by eliminating the so-called contact-stability constraints [24] on the feet. This simplification is

¹ Artificial and Mechanical Intelligence AMI (Italian Institute of Technology); Genoa, Italy {firstname.lastname}@iit.it.

² Mechanical engineering department; Politecnico di Milano, Milano, Italy {firstname.lastname}@polimi.it.

³ Machine Learning and Optimisation, The University of Manchester, Manchester, United Kingdom.

achieved thanks to a recently introduced contact-stable force parametrization [25]. Consequently, the optimization variable is changed and the number of constraints is reduced transferring the complexity to the cost function by adding terms specifically handling the payload effect. The non-linear MPC problem is solved online and validated on the example of a floating mass with two legs as well as on the humanoid robot iCub [26]. Furthermore, the computational effort required is briefly investigated specifically in terms of the effect of removing constraints from the original optimization problem through the use of the contact-stable force parametrization.

The manuscript is organized as follow; Section II introduces the notations then summarily recalls the necessary machinery used throughout this work and states the problem treated in an explicit manner. Section III introduces the proposed controller solving our problem together with some necessary comments and remarks. Section IV focuses on the validation of the proposed controller both in terms of simulations and experimental results. Finally, some concluding remarks in Section V end the manuscript.

II. BACKGROUND

In this section, we first introduce the notations used throughout, then report the basic ingredients required for designing our proposed controller together with a statement of the problem addressed.

A. Notations and nomenclature

For any vector $z \in R^n$, $\|z\|$ and z^\top define respectively the norm and transpose of z . Given a full rank matrix $B \in R^{n \times m}$ with $n > m$, $B^\dagger = (B^\top B)^{-1} B^\top$ denotes the Moore-Penrose inverse, while B^\perp its orthogonal complement verifying $B^\perp B = 0$, and $\ker\{B\}$ denotes its null space. I_n and 0_n denote the identity and zero matrices of dimension n . For $x \in R^3$, $x^\wedge = S(x) : R^3 \rightarrow so(3)$ returns the skew symmetric matrix form of x . Additionally, the following standard nomenclature will be used throughout the text;

- H^ℓ , H^ω will denote the *aggregate* linear and angular momentum of all links referred to the robot center of mass CoM and oriented as the inertial frame.
- P_{CoM} is the position of the CoM referred to the inertial frame.
- P_{C_i} , R_{C_i} , v_{C_i} will represent the position of a contact point i referred to the inertial frame, a rotation matrix expressing the orientation of a frame attached to the contact point and the linear contact velocity respectively.
- For each foot we have $n_c \in \{1, 2\}$ the total number of possible contacts at each stance phase;
- $n_v = 4$ is the number of contact vertices assuming a fixed rectangular contact surface.
- $\gamma_i \in \{0, 1\}$ is a variable capturing whether contact i is active or not.
- Finally, m is the robot mass and the gravity acceleration vector is denoted $\vec{g} = (0 \ 0 \ 9.81 \ 0 \ 0 \ 0)^\top$.

B. The centroidal momentum dynamics

The *reduced* dynamics of the centroidal quantities of a humanoid robot carrying a payload can be modeled in the standard non-linear control affine perturbed form [27];

$$\dot{x} = f(x) + G(x)u + p(x)d \quad (1)$$

where d corresponds to the external wrench(s) caused by the payload, and the drift, control and disturbance vector fields take the forms (omitting the zero block-matrix dimensions for clarity);

$$f(x) = \begin{pmatrix} f_1(x) \\ f_2(x) \\ 0 \end{pmatrix}, \quad G(x) = \begin{pmatrix} 0 & 0 \\ g_{2,1}(x) & 0 \\ 0 & g_{3,2}(x) \end{pmatrix}, \quad (2a)$$

$$p(x) = \begin{pmatrix} 0 \\ p_2(x) \\ 0 \end{pmatrix} \quad (2b)$$

where, following [2], we set

$$x = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{pmatrix} P_{CoM} \\ H \\ P_C \end{pmatrix}, \quad H = \begin{pmatrix} H^\ell \\ H^\omega \end{pmatrix} \quad (3a)$$

$$u = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} = \begin{pmatrix} f \\ v_C \end{pmatrix} \quad (3b)$$

for f, v_C vectors collecting external wrenches at the feet and contact velocities, and for each contact, capturing the contact status by letting

$$\dot{P}_{C_i} = (1 - \gamma_i)v_{C_i} \quad (4)$$

Performing the above assignment, we have explicitly;

$$\begin{aligned} f_1(x) &= \frac{1}{m} B x_2, \quad B = \begin{pmatrix} I_3 & 0_3 \end{pmatrix} \\ f_2(x) &= -m\vec{g} \\ g_{2,1}(x) &= A_f \Gamma, \quad g_{3,2}(x) = I - \Gamma \\ p_2(x) &= A_d \end{aligned}$$

where Γ is a diagonal matrix of appropriate dimensions with diagonal blocks corresponding to γ_i contacts, and the matrices

$$\begin{aligned} A_f &= (A_{f_1} \dots A_{f_{n_c \times n_v}}) \\ A_d &= (A_{d_l} \quad A_{d_r}) \\ A_{f_i} &= \begin{pmatrix} I_3 & 0_3 \\ (P_{C_i} - P_{CoM})^\wedge & I_3 \end{pmatrix} \\ A_{d_j} &= \begin{pmatrix} I_3 & 0_3 \\ (P_{d_j} - P_{CoM})^\wedge & I_3 \end{pmatrix} \end{aligned}$$

with $P_{d_j}, j \in \{l, r\}$ the contact point of the payload, assumed fixed on the left and right hands respectively. The dynamics (1) will be used for controller design in Section III.

C. The contact-stable force parametrization

For a given contact wrench;

$$w = (F_x \ F_y \ F_z \ M_x \ M_y \ M_z)^\top$$

where F_i, M_i are the corresponding force and moment in the i direction respectively, maintaining stable planar unilateral contact is tantamount to enforcing the following constraints;

$$F_z > f_z^{\min} \geq 0 \quad (5a)$$

$$\sqrt{F_x^2 + F_y^2} < \mu_c F_z \quad (5b)$$

$$y_c^{\min} < \frac{M_x}{F_z} < y_c^{\max} \quad (5c)$$

$$x_c^{\min} < -\frac{M_y}{F_z} < x_c^{\max} \quad (5d)$$

$$\left| \frac{M_z}{F_z} \right| < \mu_z \quad (5e)$$

in which $x_c^{\min}, x_c^{\max}, y_c^{\min}, y_c^{\max}$ are the dimensions of the herein assumed rectangular contact surface, and μ_c, μ_z are the static and torsional friction coefficients respectively. Constraints (5) above define a constraints set \mathcal{K} , and can be equivalently written as $w \in \mathcal{K}$.

In [25], a stable contact wrench parametrization

$$w = \phi(\xi) : R^6 \mapsto \tilde{\mathcal{K}} \subset \mathcal{K}$$

for some $\xi \in R^6$ was proposed ensuring the satisfaction of (5), where

$$\phi(\xi) = \begin{pmatrix} \mu_c \frac{\tanh \xi_1 (e^{\xi_3} + f_z^{\min})}{\sqrt{1 + \tanh^2 \xi_2}} \\ \mu_c \frac{\tanh \xi_2 (e^{\xi_3} + f_z^{\min})}{\sqrt{1 + \tanh^2 \xi_1}} \\ e^{\xi_3} + f_z^{\min} \\ (\delta_y \tanh \xi_4 + \delta_{y_0})(e^{\xi_3} + f_z^{\min}) \\ (\delta_x \tanh \xi_5 + \delta_{x_0})(e^{\xi_3} + f_z^{\min}) \\ \mu_z \tanh(\xi_6)(e^{\xi_3} + f_z^{\min}) \end{pmatrix} \quad (6)$$

and

$$\delta_x = \frac{x_c^{\max} - x_c^{\min}}{2}, \quad \delta_{x_0} = -\frac{x_c^{\max} + x_c^{\min}}{2}$$

$$\delta_y = \frac{y_c^{\max} - y_c^{\min}}{2}, \quad \delta_{y_0} = \frac{y_c^{\max} + y_c^{\min}}{2}$$

Remark 2.1: given the contact constraints (5), the parametrization (6) represents an almost global cover of the corresponding feasible contact wrenches. The constraints (5) covers our case, assuming no flight phase during walking, and a co-planar contact at the footsteps impacts, and consequently this parametrization suffices. The interested reader is referred to [25] for the technical details.

D. Problem statement

The problem we are concerned with can now be stated;

Problem 1: Given the dynamics (1), and a set of *nominal* footstep locations and corresponding CoM (and momentum) references, it is required to design a feedback control law such that the following is achieved;

- 1) **Nominal references tracking:** a bounded tracking error on both the centroidal quantities nominal references and contact locations is maintained $\forall t$.
- 2) **Disturbance attenuation:** the effect of the payload on the tracking performances is *minimized*¹.

It is imperative to note that the tracking requirement is relaxed compared to the exact tracking problem version. This is also motivated by the need to allow the feedback to provide a footstep adjustment capability. It is also clear that we are not asking, explicitly, for the feedback to respect some actuator limits, since we implicitly assume that this is taken care of by the whole-body control layer as in Figure 1. In the next section, we will detail our solution approach to this problem.

III. NON-LINEAR CENTROIDAL MPC WITH CONTACT-STABLE FORCE PARAMETRIZATION

Starting from the multi-layer architecture in Figure 1, we propose to modify the high-level trajectory adjustment layer, utilizing a receding horizon approach [28]. Indeed, based on the recently proposed online non-linear centroidal MPC controller [2], we propose to modify the prediction model, the cost function and constraints in order to solve the problem.

A. The cost function

The cost function reflects the objectives detailed in the problem statement, and to this end, the following tasks are considered in the cost function.

1) *The nominal references tracking task:* this can be further decomposed, rewriting

$$T_1 = T_h + T_{P_C} \quad (7)$$

where the centroidal dynamics task T_h reads;

$$T_h = \frac{1}{2} \|Lx_2\|_{Q_h}^2 + \frac{1}{2} \|x_1 - x_1^n\|_{Q_c}^2 \quad (8)$$

with $L = (0_3 \ I_3)$ and x_1^n a reference trajectory for the CoM position, while $Q_h, Q_c \geq 0$ positive definite penalty matrices of suitable dimension.

As for the footstep location tracking task T_{P_C} , given a nominal reference x_3^n (e.g., coming from an external planner block), we incorporate the following penalty;

$$T_{P_C} = \frac{1}{2} \|x_3 - x_3^n\|_{Q_{P_C}}^2 \quad (9)$$

in which $Q_{P_C} > 0$ is a positive definite penalizing weight.

2) *The Payload attenuation task:* this can be further decomposed into;

$$T_2 = T_d + T_\xi \quad (10)$$

where the payload attenuation (and gravity compensation) term T_d takes the form

$$T_d = \frac{1}{2} \|u_1 - (A_f)^\dagger(-A_d d) - m\bar{g} \otimes 1_{n_c \times n_v}\|_{Q_d}^2 \quad (11)$$

¹such that, barring mechanical limitations of the hardware, the robot is able to carry and walk with the payload without falling over

rewriting $u_1 = \phi(\xi)$ in terms of the parameter (our optimization variable) and $Q_d > 0$ a positive definite penalizing matrix of suitable dimension. In addition, we add a regularizing task T_ξ on the parameter value of the form;

$$T_\xi = \frac{1}{2} \|\xi\|_{Q_\xi}^2 \quad (12)$$

where $Q_\xi > 0$ is a positive definite penalizing matrix.

Remark 3.1: the regularizing task on the parameter ξ magnitude stems from the fact that

$$\lim_{\xi \rightarrow 0} \phi(\xi) = (0 \quad 0 \quad 1 + f_z^{\min} \quad 0 \quad 0 \quad 0)^\top$$

with the contact force application being in the interior of the support surface. In other words, as the parameter decreases, the corresponding contact force has no components parallel to the contact surface and are always in the interior of the feasible set.

B. The constraints

The parametrization (6) allows us to eliminate the contact-stability constraints (5) from our optimization problem. In this sense, and differently from [2], we only apply inequality constraints corresponding to restricting the maximum allowable footstep location error. Namely we set;

$$-lb \leq \|R_{C_i}(x_3^i - x_3^{n_i})\| \leq ub \quad (13)$$

where, as in (9), $x_3^{n_i}$ is a nominal reference for the i contact vertex and lb , ub are lower and upper bounds for the tracking error respectively.

C. The general MPC problem

With all the ingredients in place, we can state the general MPC problem to be solved online for $i = 1, \dots, n_p$;

$$V^* = \min \sum_{k=0}^{n_p} T_1 + T_2 \quad (14a)$$

s.t

$$x(k+i) = F(x(k+i-1), u(k+i-1)) \quad (14b)$$

$$-lb \leq \|R_{C_i}(x_3(k+i) - x_3^n(k+i))\| \leq ub \quad (14c)$$

with n_p being the so-called prediction horizon and where the prediction model (14b) is obtained by discretizing the dynamics (1) using any numerical integrator [29]. For our validation purposes, we use a simple Euler's discretization scheme. The controller outputs are generated using the Receding Horizon Principle, adopting a fixed prediction window with a length equal to n_p samples. Specifically, solving (14) for the minimizer (ξ^*, v_C^*) then the feedback solution to Problem 1 reads;

$$u^* = \left(\phi(\xi^*)^\top \quad v_C^{*\top} \right)^\top \quad (15)$$

Remark 3.2: notice that no terminal ingredients (e.g. terminal penalty and constraints set [28]) are added in this MPC problem to ensure closed loop recursive feasibility and stability. This is beyond the scope of this article.

D. The payload estimation

At any given time instant, the payload external wrench $d(t)$ and its contact location $P_{d_i}(t)$ is needed for n_p steps ahead in the prediction model to solve the optimization problem (14). This can be satisfied by having a simple dynamics modelling the evolution of both $d(t)$ and $P_{d_i}(t)$ and integrating over time. In control-theoretic terms, an *exo-system* generating the references and disturbance trajectories can be used [30]. Instead, we opt to simplify this requirement in the following manner;

Assumption 1: For $t = k\delta$, δ being the sampling period used to integrate the prediction model, and given an *estimate* of the payload external wrench $\hat{d}(k)$ the following holds;

$$d(k+i) = \hat{d}(k), \quad i = 1 \dots n_p$$

Moreover, we assume the payload contact location w.r.t. CoM to be given by $P_{d_r}(k+i) = \bar{P}_{d_r}(k)$, $P_{d_l}(k+i) = \bar{P}_{d_l}(k)$ for $i = 1 \dots n_p$ where $\bar{P}_{d_r}(k)$, $\bar{P}_{d_l}(k)$ are the right and left hands positions at time k respectively.

It is worth stressing that this is a valid assumption since the payload disturbance is updated in the next iteration of the MPC thanks to the receding horizon implementation.

When testing on the robot, both $\hat{d}(k)$ and $\bar{P}_{d_i}(k)$ are needed. For the payload contact location, the robot kinematics model is enough to obtain, for each time step, the values of \bar{P}_{d_i} . As for the payload external wrench, we use an estimation algorithm that relies on propagating the measurements obtained by a given set of force-torque sensors through the kinematic chain, writing and aggregating the Newton's-Euler's equations for each rigid link in the sub-chain and solving for the external wrench via a least-square norm-minimization-like technique (for more details the reader is referred to [31] and the references therein).

To conclude this section, some remarks are in order:

- the task (11) results from applying direct disturbance compensation techniques. When in double support, there may exist infinitely many values for the parameter ξ satisfying the requirement. A projection on the null space to enforce a given selection of the minimizer may be used [25]. The current set-up however does not enforce a strict prioritization of (11).
- neglecting the disturbance compensation (11) and parameter regularization (12) tasks, the MPC problem (14) is not necessarily equivalent to that in Section III-D of [2]. Consequently, as explained in Remark 2.1, there exist feasible feet contact wrenches solving the original problem, for which no optimum parameter solving (14) can be obtained. However, for our purposes, the parametrization covers almost all (90% of) the feasible forces space [25].

IV. RESULTS AND VALIDATION

In this section, we validate the proposed controller on several scenarios and comment on the obtained results. In all cases, we use Casadi[®] [32], IPOPT [33] and the linear solver HSL MA97[®] [34] to solve the optimization problem

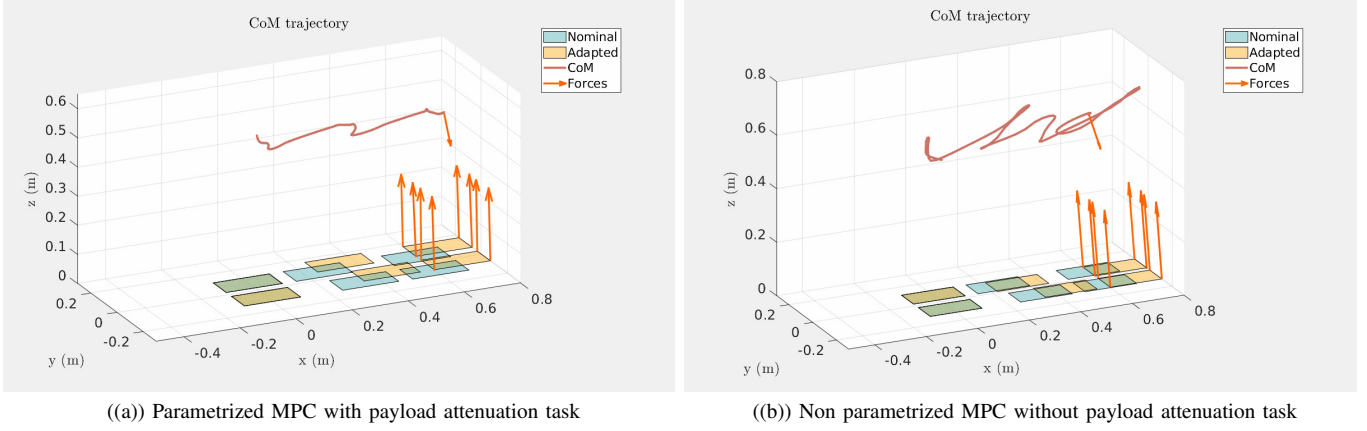


Fig. 2: Comparison of obtained CoM trajectories and adapted contact locations of the proposed parametrized MPC with payload attenuation (a), and a non-linear centroidal MPC without payload attenuation (b). The payload external force acting on the CoM is also depicted with an orange arrow, together with the feet contact forces.

n_p [time steps]	10	MPC rate [s]	0.2
μ_c [N/A]	0.33	f_z^{\min}	0.01
Q_h	$10^2 \times I_3$	Q_c	$\text{diag}\{1, 1, 10^3\}$
Q_{P_c}	$200 \times I_3$	Q_d	$10^2 \times I_3$
Q_ξ	$10 \times I$		

TABLE I: parameters and penalizing weights

(14). In addition, table I list the various parameters and weight penalties that are fixed throughout the validation case studies. The proposed controller runs on a 10th generation Intel® Core i7-10750H laptop equipped with Ubuntu Linux 20.04. For bench-marking reasons, and whenever mentioned, we compare the modified payload-aware controller against the MPC controller presented in [2]. This is done using parameters and penalizing weights that were originally tuned for the controller in [2].

A. Reduced models simulations

We first validate the proposed approach in the case of a floating mass equipped with two legs. In this case we set $m = 1$ kg, $n_c \in \{1, 2\}$, $n_v = 4$ and the CoM height being $\bar{c}_z = 0.53$ m. The system is subject to a 1.5 kg payload at an assumed fixed payload contact location being $P_{d_i} = (0.25 \pm 0.1 \quad -\frac{\bar{c}_z}{4})^\top$ for the left and right payload contact location w.r.t. to the CoM respectively. Figure 2 depicts the generated CoM trajectories and foot-step locations and contact forces in this case.

It is clear from the simulation figures that including an the payload attenuation task in the MPC problem results in more reasonable CoM trajectory, i.e. with less variation in the CoM height for the same tuning parameters. More interestingly, while the feet contact forces are comparable in both cases, the proposed parametrized centroidal MPC compensates for the payload with larger steps adaptation.

B. Validation on the iCub3 robot

To test the proposed controller performances on a humanoid robot, we use the iCub3 [35], a 1.25 m tall, and weighs 52 kg. The tests were carried first in simulation

using Gazebo [36] and then replicated on the real hardware (modulo sensor/actuator noise and wrist mechanical construction limits). Indeed, in this case, the payload contact locations are obtained online at each time step using the robot kinematics assuming a fixed contact location at the hand reference frame. Additionally, and as is done in [2], the whole-body controller is a Quadratic Program - QP over the robot generalized velocity and the joint velocities. The QP program is solved using an off the shelf solver.

In this scenario, we give the robot planned nominal feet steps and corresponding trajectories. In addition, the robot is carrying a box that contains a small weight of 2 kg (due to the iCub3 wrist construction limitations²). Figure 3 illustrates the obtained performances in terms of tracking of the CoM and the feet contact locations adaptation. As expected, the CoM tracking error is about 0.01 m – 0.04 m with persistent adaptations on the feet tracking to compensate for the payload (i.e with a peak of about 0.021 m). The robot manages to conclude the task successfully.

C. Comments on computational aspects

One interesting question concerning the simplification of the MPC problem by using the parametrization (6) to reduce the number of constraints would be; *what are the gains, if any, in terms of computation time and number of iterations?*

To fairly judge this aspect, we removed the corresponding payload attenuation (11) and parameter regularization (12) tasks. Those instead were replaced with a force regularization term rendering the feet contact forces as similar as possible. In this sense, we are able to compare the parametrized centroidal MPC against the original version developed in [2]. Figure 4 illustrates the computational time in both problems in one run where the objective is to allow the robot to walk following some nominal planned footstep trajectory.

²In simulations, as explained in the accompanying video, we are able to attach a mass of $m = 10$ kg. Related code https://github.com/ami-iit/paper_elobaid_2023_icra_walking_with_payloads.git

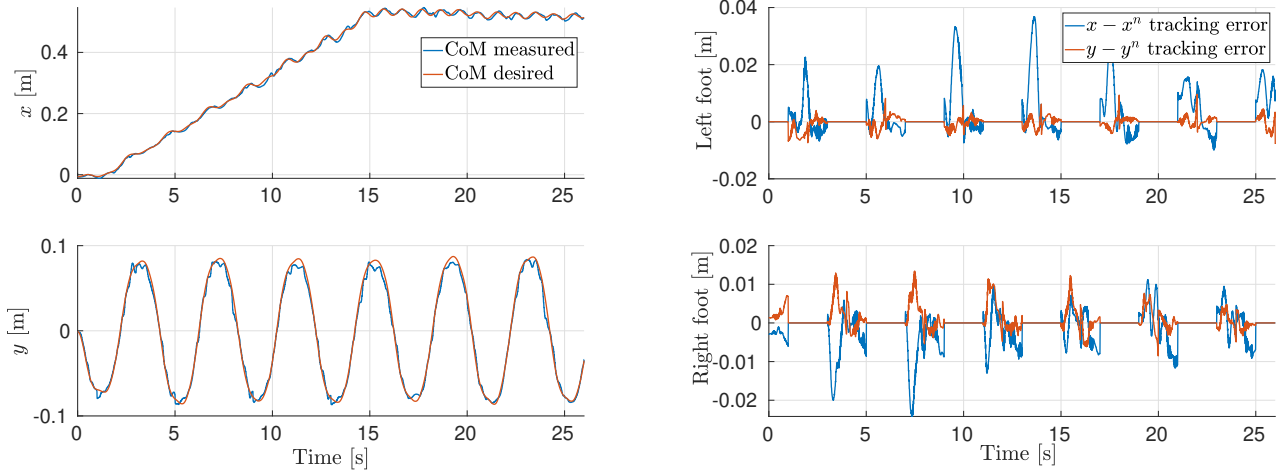
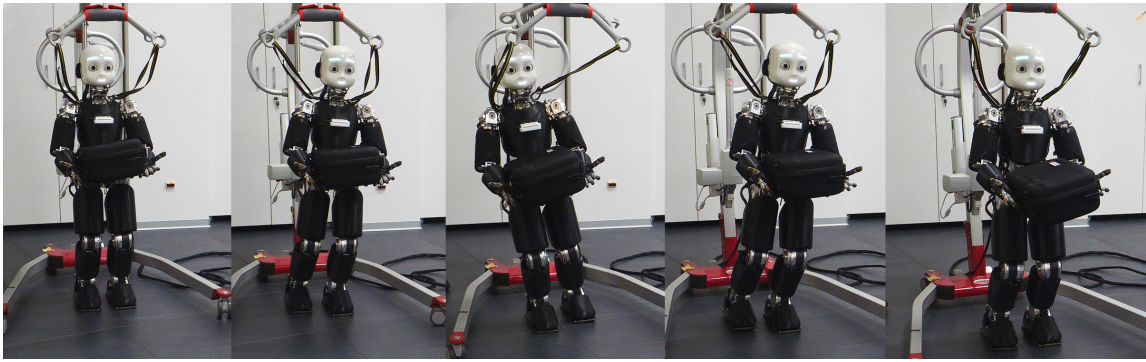


Fig. 3: Top: screenshots of the robot walking while carrying the payload. Bottom left: the CoM tracking in closed loop. Bottom right: the tracking errors of the generated (adapted) feet contact locations w.r.t the nominal ones.

This direct comparison shows that while in a lot of time steps, the original centroidal MPC out-performs the proposed controller in terms of computation time required, however on average, the proposed controller gives better results (0.068 s compared to 0.073 s). It is also clear that the performance gain is not by much. In essence, removal of constraints in a non-linear optimization problems does not necessarily lead to *significant*, if any, computational gains.

V. CONCLUSIONS AND PERSPECTIVES

This work presented an online non-linear centroidal MPC controller in combination with an external payloads estimates and a stable-force parametrization to allow a humanoid robot to carry objects while walking. The “robustness” of the proposed controller stems from a specifically designed term in the cost function that rejects the effect of the disturbance, as well as including the payload effects on the centroidal quantities dynamics in the prediction model. The controller is validated both in simulations and on a newer version of the humanoid robot iCub. Being a simplified MPC controller thanks to the force parametrization, some computational gains are obtained, albeit being rather small.

As a future direction of research, we are investigating additional avenues of lowering the computational time required. In addition, we are interested in closing the loop on the contact forces to achieve better contact forces tracking.

ACKNOWLEDGMENT

This work was partially supported by the Italian National Institute for Insurance against Accidents at Work (INAIL) ergoCub Project, and Honda Research and Development Japan, through a joint-lab research initiative.

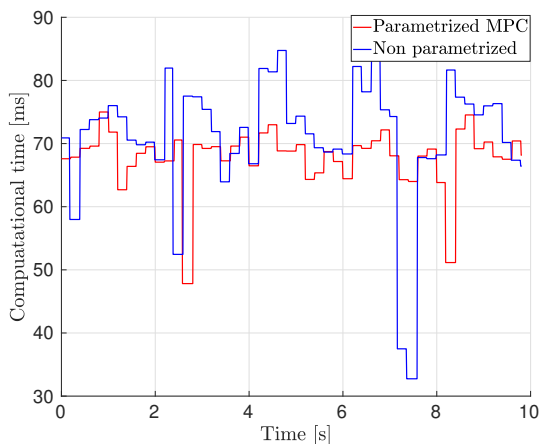


Fig. 4: Computational time comparison

REFERENCES

- [1] T. Koolen, S. Bertrand, G. Thomas, T. De Boer, T. Wu, J. Smith, J. Engelsberger, and J. Pratt, "Design of a momentum-based control framework and application to the humanoid robot atlas," *International Journal of Humanoid Robotics*, vol. 13, no. 01, p. 1650007, 2016.
- [2] 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 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 10412–10419.
- [3] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa, "The 3d linear inverted pendulum mode: a simple modeling for a biped walking pattern generation," in *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the Next Millennium (Cat. No.01CH37180)*, vol. 1, 2001, pp. 239–246 vol.1.
- [4] G. Romualdi, S. Dafarra, Y. Hu, and D. Pucci, "A benchmarking of dcm based architectures for position and velocity controlled walking of humanoid robots," in *2018 IEEE-RAS 18th International Conference on Humanoid Robots (Humanoids)*, 2018, pp. 1–9.
- [5] J. Engelsberger, C. Ott, and A. Albu-Schäffer, "Three-dimensional bipedal walking control using divergent component of motion," in *2013 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2013, pp. 2600–2607.
- [6] R. J. Griffin, G. Wiedebach, S. Bertrand, A. Leonessa, and J. Pratt, "Walking stabilization using step timing and location adjustment on the humanoid robot atlas," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2017, pp. 667–673.
- [7] N. Ramuzat, G. Buondonno, S. Boria, and O. Stasse, "Comparison of position and torque whole-body control schemes on the humanoid robot talos," in *2021 20th International Conference on Advanced Robotics (ICAR)*, 2021, pp. 785–792.
- [8] D. E. Orin and A. Goswami, "Centroidal momentum matrix of a humanoid robot: Structure and properties," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 653–659.
- [9] L. Rapetti, Y. Tirupachuri, A. Ranavolo, T. Kawakami, T. Yoshiike, and D. Pucci, "Shared control of robot-robot collaborative lifting with agent postural and force ergonomic optimization," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 9840–9847.
- [10] P.-b. Wieber, "Trajectory free linear model predictive control for stable walking in the presence of strong perturbations," in *2006 6th IEEE-RAS International Conference on Humanoid Robots*, 2006, pp. 137–142.
- [11] R. J. Griffin, A. Leonessa, and A. Asbeck, "Disturbance compensation and step optimization for push recovery," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 5385–5390.
- [12] J. W. Grizzle, C. Chevallereau, R. W. Sinnet, and A. D. Ames, "Models, feedback control, and open problems of 3d bipedal robotic walking," *Automatica*, vol. 50, no. 8, pp. 1955–1988, 2014.
- [13] B. Stephens, "Humanoid push recovery," in *2007 7th IEEE-RAS International Conference on Humanoid Robots*, 2007, pp. 589–595.
- [14] H. Jeong, I. Lee, J. Oh, K. K. Lee, and J.-H. Oh, "A robust walking controller based on online optimization of ankle, hip, and stepping strategies," *IEEE Transactions on Robotics*, vol. 35, no. 6, pp. 1367–1386, 2019.
- [15] K. Harada, S. Kajita, H. Saito, M. Morisawa, F. Kanehiro, K. Fujiwara, K. Kaneko, and H. Hirukawa, "A humanoid robot carrying a heavy object," in *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*. IEEE, 2005, pp. 1712–1717.
- [16] K. Harada, S. Kajita, F. Kanehiro, K. Fujiwara, K. Kaneko, K. Yokoi, and H. Hirukawa, "Real-time planning of humanoid robot's gait for force controlled manipulation," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, vol. 1, 2004, pp. 616–622 Vol.1.
- [17] H. Arisumi, J.-R. Chardonnet, A. Kheddar, and K. Yokoi, "Dynamic lifting motion of humanoid robots," in *Proceedings 2007 IEEE International Conference on Robotics and Automation*, 2007, pp. 2661–2667.
- [18] D. J. Agravante, A. Sherikov, P.-B. Wieber, A. Cherubini, and A. Kheddar, "Walking pattern generators designed for physical collaboration," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, 2016, pp. 1573–1578.
- [19] M. V. Minniti, R. Grandia, F. Farshidian, and M. Hutter, "Adaptive clf-mpc with application to quadrupedal robots," *IEEE Robotics and Automation Letters*, vol. 7, no. 1, pp. 565–572, 2021.
- [20] H. Diedam, D. Dimitrov, P.-B. Wieber, K. Mombaur, and M. Diehl, "Online walking gait generation with adaptive foot positioning through linear model predictive control," in *2008 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2008, pp. 1121–1126.
- [21] N. A. Villa and P.-B. Wieber, "Model predictive control of biped walking with bounded uncertainties," in *2017 IEEE-RAS 17th International Conference on Humanoid Robotics (Humanoids)*, 2017, pp. 836–841.
- [22] F. M. Smaldone, N. Scianca, V. Modugno, L. Lanari, and G. Oriolo, "Gait generation using intrinsically stable mpc in the presence of persistent disturbances," in *2019 IEEE-RAS 19th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2019, pp. 651–656.
- [23] K. Kaneko, F. Kanehiro, M. Morisawa, E. Yoshida, and J.-P. Laumond, "Disturbance observer that estimates external force acting on humanoid robots," in *2012 12th IEEE International Workshop on Advanced Motion Control (AMC)*, 2012, pp. 1–6.
- [24] R. Featherstone, *Rigid body dynamics algorithms*. Springer, 2014.
- [25] A. Gazar, G. Nava, F. J. A. Chavez, and D. Pucci, "Jerk control of floating base systems with contact-stable parameterized force feedback," *IEEE Transactions on Robotics*, vol. 37, no. 1, pp. 1–15, 2021.
- [26] G. Metta, L. Natale, F. Nori, G. Sandini, D. Vernon, L. Fadiga, C. Von Hofsten, K. Rosander, M. Lopes, J. Santos-Victor, et al., "The icub humanoid robot: An open-systems platform for research in cognitive development," *Neural networks*, vol. 23, no. 8-9, pp. 1125–1134, 2010.
- [27] A. Isidori, *Nonlinear Control Systems*. Springer-Verlag, 1995.
- [28] L. Grüne and J. Pannek, "Nonlinear model predictive control," in *Nonlinear model predictive control*. Springer, 2017, pp. 45–69.
- [29] G. F. Franklin, J. D. Powell, M. L. Workman, et al., *Digital control of dynamic systems*. Addison-wesley Reading, MA, 1998, vol. 3.
- [30] C. Byrnes and A. Isidori, "Output regulation for nonlinear systems: an overview," *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, vol. 10, no. 5, pp. 323–337, 2000.
- [31] S. Ivaldi, M. Fumagalli, M. Randazzo, F. Nori, G. Metta, and G. Sandini, "Computing robot internal/external wrenches by means of inertial, tactile and f/t sensors: Theory and implementation on the icub," in *2011 11th IEEE-RAS International Conference on Humanoid Robots*, 2011, pp. 521–528.
- [32] J. A. E. Andersson, J. Gillis, G. Horn, J. B. Rawlings, and M. Diehl, "CasADi – A software framework for nonlinear optimization and optimal control," *Mathematical Programming Computation*, vol. 11, no. 1, pp. 1–36, 2019.
- [33] A. Wächter and L. T. Biegler, "On the implementation of an interior-point filter line-search algorithm for large-scale nonlinear programming," *Mathematical programming*, vol. 106, no. 1, pp. 25–57, 2006.
- [34] H. S. Library. Hsl for ipopt. [Online]. Available: <https://www.hsl.rl.ac.uk/ipopt/>
- [35] S. Dafarra, K. Darvish, R. Grieco, G. Milani, U. Pattacini, L. Rapetti, G. Romualdi, M. Salvi, A. Scalzo, I. Sorrentino, et al., "icub3 avatar system," *arXiv preprint arXiv:2203.06972*, 2022.
- [36] N. Koenig and A. Howard, "Design and use paradigms for gazebo, an open-source multi-robot simulator," in *2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566)*, vol. 3. IEEE, 2004, pp. 2149–2154.