Adaptive Non Linear Centroidal MPC ensuring Robust Locomotion for legged robots

Laganà Lorenzo 1937829 , Daniele Teni 2115180 , Tran Hoan An Nguyen 2113021

February 2025

1 Introduction

A simplified model in order to study the Locomotion of a legged robot is the LIP (Linear Inverted Pendulum) in which the dynamics of the robot is described as an inverted pendulum in which the base is given by the the ZMP and the top is given by the CoM of the robot. It turns out that the ZMP pushes the CoM away (i.e. the acceleration of the CoM is directed in the direction of falling for the pendulum).
This model extremely simplifies the dynamics and it will be not usefull for some applications for example to compensate a disturbance given by a payload carried out by the robot.
For this reason we want to investigate another model for the dynamics of the robot.

2 Centroidal Momentum Dynamics

Consider an inertial frame or *world frame* w. We assume that the robot is composed of n+1 rigid bodies(links) connected by **n** joints with one degree of freedom each.

We can characterize the robot configuration space by the *position* and the *orientation* of a frame attached to one robot's link (in generale the torso) that is called **Base frame** $\mathcal{B}$ and the *joint configurations*. So $\mathbb{Q} \in \mathbb{R}^3 \times SO(3) \times \mathbb{R}^n$. An element of $\mathbb{Q}$ is a triplet $q = ({}^w p_{\mathcal{B}}, {}^w R_{\mathcal{B}}, q_j)$ where $({}^w p_{\mathcal{B}}, {}^w R_{\mathcal{B}})$ denotes the origin position and orientation of $\mathcal{B}$ expressed with respect to the world frame w, while q_j denotes the joint angles.

Let's consider the full dynamics of a legged robot, taking into account $\mathbf{n_c}$ environment exchanging distinct wrenches (always present due to the ground reaction force):

$$M(q)\dot{v} + C(q,v)v + G(q) = B\tau + \sum_{k=1}^{n_c} J_k^T f_k \tag{1}$$

where: $v = ({}^w\dot{p}_{\mathcal{B}}, {}^w\omega_{\mathcal{B}}, \dot{q}_j) \in \mathbb{V} = \mathbb{R}^3 \times \mathbb{R}^3 \times \mathbb{R}^n$ is the generalized robot's velocity, with ${}^w\dot{p}_{\mathcal{B}}$ linear velocity of the base frame, and ${}^w\omega_{\mathcal{B}} = {}^w\dot{R}_{\mathcal{B}} = S({}^w\omega_{\mathcal{B}})\,{}^wR_{\mathcal{B}}$ angular velocity of the base frame with respect to the world frame w.
$M \in \mathbb{R}^{n+6 \times n+6}$ is the inertia matrix, $C \in \mathbb{R}^{n+6 \times n+6}$ is the Coriolis matrix, $G \in \mathbb{R}^{n+6}$ are the gravity terms and $b_1 = [0_{6\times 6}, 0_{n\times 6}]^T$, $b_2 = [0_{6\times n}, I_{n\times n}]^T$ so $B = [b_1, b_2]$ is the selector matrix (the base frame is not actuated).

Moreover $\tau \in \mathbb{R}^n$ is the torques vector of the actuated joints, and $f_k \in \mathbb{R}^6$ denotes the *k-th* external wrench applied by the environment to the robot in the *k-th* contact point.

These forces are expressed in a frame F_k oriented in the same way as the Base frame but the origin is located at the point of application of the force. So the jacobian $J_k = J_k(q)$ is the map between the robot's velocity v and the linear and angular velocity ${}^w v_{F_k} := ({}^w\dot{p}_{F_k}, {}^w\omega_{F_k})$ of the frame F_k with respect to the base frame i.e. ${}^w v_{F_k} = J_{F_k}(q)v$.
It has the following structure:

$$J_{C_k}(q) = \begin{bmatrix} J^b_{C_k}(q) & J^j_{C_k}(q) \end{bmatrix} \in \mathbb{R}^{6\times(n+6)}, \qquad J^b_{C_k}(q) = \begin{bmatrix} I_{3\times3} & -S(p_{F_k} - p_{CoM}) \\ \mathbf{0}_{3\times3} & I_{3\times3} \end{bmatrix} \in \mathbb{R}^{6\times6}.$$

Remark Note that $S(x) \in \mathbb{R}^{3\times3}$ be the skew-symmetric matrix such that $S(x)y = x \times y$, where $\times$ is the cross product operator in $\mathbb{R}^3$.

Now following [1],[2] we can consider a coordinate trasformation in the state (q,v) that puts the system dynamics (1) into a new form in which the matrix M is block diagonal.

This means that we can decouple the joint and base fras accelerations.
Consider the following partition of the inertia matrix:

$$M = \begin{bmatrix} M_b & M_{bj} \\ M_{jb} & M_j \end{bmatrix} \qquad \text{with } M_b \in \mathbb{R}^{6\times6} \quad , \quad M_{bj} \in \mathbb{R}^{6\times n} \quad , \quad M_j \in \mathbb{R}^{n\times n}$$

and the frame c with origin in the CoM of the robot and with the same orientation of w, we can define this coordinate trasformation:

$$q := q \qquad \bar{v} := T(q)v$$

where

$$T := \begin{bmatrix} {}^cX_{\mathcal{B}} & {}^cX_{\mathcal{B}} M_b^{-1} M_{bj} \\ 0_{n\times6} & I_{n\times n} \end{bmatrix} \qquad {}^cX_{\mathcal{B}} := \begin{bmatrix} I_{3\times3} & -S(p_{F_k} - p_{CoM}) \\ \mathbf{0}_{3\times3} & I_{3\times3} \end{bmatrix}$$

Remark The matrix T cannot be singular, it can always be inverted.

The new dynamics in $(q, \bar{v})$ is given by:

$$\bar{M}(q)\dot{\bar{v}} + \bar{C}(q, \bar{v})\bar{v} + \bar{G}(q) = B\tau + \sum_{k=1}^{n_c} \bar{J}_k^T f_k \tag{2}$$

with:

$$\bar{M}(q) = T^{-\top} M T^{-1} = \begin{bmatrix} \bar{M}_b(q) & 0_{6\times n} \\ 0_{n\times6} & \bar{M}_j(q_j) \end{bmatrix}, \qquad \bar{C}(q, \bar{\nu}) = T^{-\top}\left(M\dot{T}^{-1} + CT^{-1}\right),$$

$$\bar{G} = T^{-\top} G = mge_3, \qquad \bar{J}_{F_i}(q) = J_{F_i}(q)T^{-1} = \begin{bmatrix} \bar{J}^b_{F_i}(q) & \bar{J}^j_{F_i}(q_j) \end{bmatrix},$$

$$\bar{M}_b(q) = \begin{bmatrix} mI_3 & 0_{3\times3} \\ 0_{3\times3} & \mathbb{I}(q) \end{bmatrix}, \quad \bar{J}^b_{F_i}(q) = \begin{bmatrix} I_3 & -S(p_{F_i} - {}^w p_c) \\ 0_{3\times3} & I_3 \end{bmatrix}.$$

Where $\mathbb{I}(q)$ is the inertia matrix computed with respect to the CoM frame c, oriented as w, and $e_3 \in \mathbb{R}^{n+6}$ is the third canonical base vector.

As we can see $\bar{M}$ is block diagonal and so the transformed base acceleration is indipendent from the joint acceleration.

More precisely the transformed robot generalized velocity is given by $\bar{v} = ({}^w\dot{p}_c^T, {}^w\omega_c^T, \dot{q}_j^T)^T$ where ${}^w\dot{p}_c$. is the velocity of the center-of-mass of the robot, and ${}^w\omega_c$ is the so-called *average angular velocity.*

In particular the equation (2) unifies what in literature is presented with two set of equations: the upper part is the so-called **Centroidal dynamics** when it is expressed in terms of *average angular velocity*, while the bottom part is the *free floating system dynamics.*

3 Control of the Centroidal Momentum Dynamics

The previous section puts in light the **Centroidal dynamics**, i.e. the rate of change of the robot's momentum expressed at the CoM, which then equals the summation of all the external wrenches acting on the multibody system [3], in the present case reduces to the contact wrench f plus the gravity wrench. These are the equation that describes the motion of a rigid body with mass m and variable inertia.

The *robot's momentum* $h = (h_\ell^T, h_\omega^T)^T \in \mathbb{R}^6$ is given by $h = \bar{M}_b v_{\mathcal{B}} = \bar{M}_b * ({}^w\dot{p}_{\mathcal{B}}^T, {}^w\omega_{\mathcal{B}}^T)^T$. The centroidal dynamics is exactly the first equation of (2) and we can write it as:

$$\dot{h} = \frac{d}{dt}(M_b v_{\mathcal{B}}) = J_b^T f - mge_3 = \sum_{k=1}^{n_c} J_{b_k}^T f_k - mge_3 = \sum_{k=1}^{n_c} \begin{bmatrix} I_3 & 0 \\ -S({}^w p_{F_k} - {}^w p_c)^T & I_3 \end{bmatrix} f_k - mge_3$$

Now we need to consider a main **assumption**:

- For a foot in contact with the ground, we can imagine that the exchanged forces are applied to the vertices of the rectangula foot, and in that way we can neglect the momentum in the exchanged wrench, so it contains only linear forces, i.e. $f \in \mathbb{R}^3$.
- To control the robot momentum, it is assumed that the contact impact forces can be chosen at will, so they can be seen as our **control inputs** $u \in \mathbb{R}^3$

So the dynamics of the robot's momentum can be expres in that way:

$$\dot{h} = J_b^T u - mge_3 = \sum_{k=1}^{n_c} \begin{bmatrix} I_3 \\ S({}^w p_{u_k} - {}^w p_c) \end{bmatrix} u_k - mge_3 \quad = \quad \sum_{k=1}^{n_c} A_k(p_{u_k})\Gamma_k u_k + m\overrightarrow{g}$$

Where $\Gamma_k = 1, 0$ is a variable that captures the status of the contact k. Now in order to have a full rapresentation of the centroidal quantities dynamics we consider also the evolution of the position of the CoM, its dynamics is simply given by $\dot{p}_{CoM} = \frac{1}{m}Bh$ where $B = [I_3, 0]$ is a selector matrix.

So we will consider this reduced dynamics to study the behaviour of the robot:

$$\begin{cases} \dot{p}_{CoM} = \frac{1}{m}Bh, \\ \dot{h} = \sum_{k=1}^{n_c} A_k(p_{u_k})\Gamma_k u_k + m\overrightarrow{g}. \end{cases} \tag{3}$$

That is the union of the centroidal momentum dynamics and the CoM dynamics. *This is not an approximate model, it is exact and gives the relation between forces and macroscopit rappresentation of the robot.*

4 Centroidal dynamics Control in presence of disturbances

Let's consider now a **constant external disturbance force** $\theta \in \mathbb{R}^3$, we consider to have i distrubance acting on the system at the points p_{θ_i}. Following the same reasoning of before we can write the momentum dynamics as:

$$\dot{h} = J_b^T u + J_b^T \theta - mge_3 = \sum_{k=1}^{n_c} A_k(p_{u_k})\Gamma_k u_k + \sum_{k=1}^{i} A_k(p_{\theta_k})\theta_k + m\overrightarrow{g}$$

We're considering that the robot is carrying out a non-changing payload with its hands. For sake of completeness we report the whole dynamics with and without a distinction between $\dot{h}_\ell$ and $\dot{h}_\omega$:

$$\begin{aligned} \dot{p}_{CoM} &= \frac{1}{m}Bh, \\ \dot{h} &= \sum_{k=1}^{n_c} A(p_{u_k})\Gamma_k u_k + \sum_{k=1}^{i} A(p_{\theta_k})\theta_k + m\overrightarrow{g} \end{aligned} \quad \overset{\text{or}}{\longrightarrow} \quad \begin{aligned} \dot{p}_{CoM} &= \frac{1}{m}Bh = \frac{1}{m}h_\ell \\ \dot{h}_\ell &= \sum_{k=1}^{n_c} \Gamma_k u_k + \sum_{k=1}^{i} \theta_k + m\overrightarrow{g} \\ \dot{h}_\omega &= \sum_{k=1}^{n_c} S(p_{u_k} - p_c)\Gamma_k u_k + \sum_{k=1}^{i} S(p_{\theta_k} - p_c)\theta_k \end{aligned} \tag{4}$$

5 Parametric-pure feedback form

Let's now put the system in a form for which we have a lot of tools in the control literature. Our goal now is to put the system in a SISO control affine form:

$$\begin{aligned} \dot{x} &= f_0(x) + \sum_{k=1}^{p} \theta_k f_k(x) + \left[g_0(x) + \sum_{k=1}^{p} \theta_k g_k(x) \right] u, \\ y &= h(x). \end{aligned}$$

With $x = (P_{CoM}^T, h^T)^T \in \mathbb{R}^n$ and $f(x), g(x)$ complete vector fields and $h(x)$ being the output map, with the unknown parameters θ in a compact convex set. Moreover we consider a well defined relative degree r.

We can define a θ-dependent coordinate trasformation $\phi(x) : (x) \to (\xi, \eta)$ such that in the new coordinates the system possesses a parametric-pure feedback form[4] (or Brunoski canonical form):

$$\begin{aligned} \dot{\xi}_1 &= \xi_2 + \theta^\top \alpha_1(\xi_1, \xi_2, \eta) \\ \dot{\xi}_2 &= \xi_3 + \theta^\top \alpha_2(\xi_1, \xi_2, \xi_3, \eta) \\ &\vdots \\ \dot{\xi}_{r-1} &= \xi_r + \theta^\top \alpha_{r-1}(\xi_1, \dots, \xi_r, \eta) \\ \dot{\xi}_r &= \alpha_0(\xi, \eta) + \theta^\top \alpha_r(\xi_r) + \beta_0(\xi, \eta)u \\ \eta &= q_0(\xi, \eta) + \sum_{i=1}^{p} \theta_i q_i(\xi, \eta) \\ y &= \xi_1 \end{aligned} \quad \begin{gathered} \longrightarrow \\ \text{In our case we} \\ \text{can select} \\ (\xi_1, \xi_2, \eta)^T = \\ (P_{CoM}, h_\ell, h_\omega), \\ \text{getting:} \\ \longrightarrow \end{gathered} \quad \begin{aligned} \dot{\xi}_1 &= \frac{1}{m}\xi_2 \\ \dot{\xi}_2 &= \sum_{k=1}^{n_c} \Gamma_k u_k + \sum_{k=1}^{i} \theta_k + m\overrightarrow{g} \\ \dot{\eta} &= \sum_{k=1}^{n_c} S(p_{u_k} - p_c)\Gamma_k u_k + \sum_{k=1}^{i} S(p_{\theta_k} - p_c)\theta_k \\ y &= \xi_1 \end{aligned}$$

So the perturbed centroidal dynamics is transformed into a parametric-pure feedback form.

6 Problem statement

Given the following assumption:

- We have a *nominal reference* $p^d_{CoM}(t)$ for the center of mass and its first and second order derivative,
- We consider that m=1 and $n_c = i = 1$ for sake of simplicity,
- The disturbance acting on the CoM is constant.

Then we want to designa *control input* u such that:

- For a bounded distrubance θ in the ℓ_2 norm then the tracking error remains bounded, specifically: $\lim_{t\to\infty} ||p_{CoM} - p^d_{CoM}|| \le \epsilon$ for some $\epsilon \in \mathbb{R}$
- The closed loop system is *Lyapuno stable*, i.e. the solution trajectories of the closed-loop system remain near an equilibrium for all time. (not requiring of prfect tracking)

7 Robust Control law: Robust Adaptive Redesign approach

We want to consider the Robust Adaptive Redesign tecnique[4] that is very usefull to get a robust control law with respect to the distrubances.

Let us define the following coordinates change that is a slightly variation of the previous trasformation needed to put the system in parametric-pure feedback form, in fact z_1 rappresents the error of the p_{CoM} and z_2 is pratically the error in $\dot{p}_{CoM}$ plus some term to simplify the reuslt and η mantain the same meaning:

$$\begin{aligned} z_1 &= p_{CoM} - p^d_{CoM} \\ z_2 &= k_1(p_{CoM} - p^d_{CoM}) + Bh - \dot{p}^d_{CoM} \quad = \quad k_1 z_1 + Bh - \dot{p}^d_{CoM} \\ \eta &= Ch. \end{aligned} \tag{5}$$

Where k_1 is a suitable diagonale *gain matrix*, while the matrix C is such taht $\eta = Ch = h_\omega$ so $C = [0_3\ ,\ I_3]$.

We apply it to the system (4) and we end up with a parametric-pure feedback form as well. In these new coordinates the dynamics is given by:

$$\begin{aligned} \dot{\boldsymbol{z}}_{\boldsymbol{1}} &= Bh - \dot{p}^d_{CoM} \quad = \quad BB^T(-k_1 z_1 + z_2 + \dot{p}^d_{CoM}) - \dot{p}^d_{CoM} \quad = \quad \boldsymbol{-k_1 z_1 + z_2} \\ \dot{\boldsymbol{z}}_{\boldsymbol{2}} &= k_1 \dot{z}_1 + B\dot{h} - \ddot{p}^d_{CoM} \quad = \quad \boldsymbol{-k_1^2 z_1 + k_1 z_2 + B\overrightarrow{g} + u + \theta - \ddot{p}^d_{CoM}} \\ \dot{\boldsymbol{\eta}} &= C\dot{h} \quad = \quad \boldsymbol{S(p_u - p_{CoM})u + S(p_\theta - p_{CoM})\theta} \end{aligned}$$

Note that $BB^T = I_3$. We treat the disturbance θ as an *unknown* variable, so we need to introduce the *disturbance estimation* $\hat{\theta}$ and we want *adaptively update* this estimation throught the dynamics of the system. For this reason we introduce also the *disturbance estimation error* $\tilde{\theta} = \theta - \hat{\theta}$.
Now consider a suitable gain matrix k_2 to be tuned and we can define the control input:

$$\boldsymbol{u_{tracking} = u_t = -(k_1 + k_2)z_2 + k_1^2 z_1 - B\overrightarrow{g} - \hat{\theta} + \ddot{p}^d_{CoM}} \tag{6}$$

The z-sub dynamics becames:

$$\begin{aligned}
\dot{\boldsymbol{z}}_{\boldsymbol{1}} &= -\boldsymbol{k_1 z_1} + \boldsymbol{z_2} \\
\dot{\boldsymbol{z}}_{\boldsymbol{2}} &= -k_1^2 z_1 + k_1 z_2 + B\overrightarrow{g} + (-(k_1+k_2)z_2 + k_1^2 z_1 - B\overrightarrow{g} - \hat{\theta} + \ddot{p}_{CoM}^d) + \theta - \ddot{p}_{CoM}^d \\
&= -k_2 z_2 + (\theta - \hat{\theta}) \quad = \quad \boldsymbol{-k_2 z_2 + \tilde{\theta}}
\end{aligned}$$

Following [4] the chosen adaptive law for the estimate is given by:

$$\dot{\hat{\theta}} = z_2$$

And this make sense because, if the robot is tracking the trajectory discarding the disturbance, then $z_1 \to 0$ and $z_2 \to 0$. But since we have a disturbance θ that acts on the z_2 dynamics it cannot be 0, so we can measure z_2 and feed it back to compensate this disturbance, that's exactly what we do with our controller.
The Whole augmented dynamics is given by:

$$\begin{aligned}
\dot{\boldsymbol{z}}_{\boldsymbol{1}} &= -\boldsymbol{k_1 z_1} + \boldsymbol{z_2} \\
\dot{\boldsymbol{z}}_{\boldsymbol{2}} &= -\boldsymbol{k_2 z_2} + \boldsymbol{\tilde{\theta}} \\
\dot{\hat{\boldsymbol{\theta}}} &= \boldsymbol{z_2} \\
\dot{\boldsymbol{\eta}} &= \boldsymbol{S(p_u - p_{CoM})\big(-(k_1+k_2)z_2 + k_1^2 z_1 - B\overrightarrow{g} - \hat{\theta} + \ddot{p}_{CoM}^d\big) + S(p_\theta - p_{CoM})\theta}
\end{aligned} \tag{7}$$

7.1 Stability

Let's study the *stability of the origin* throught a Lyapunov analisys. We consider the following candidate Lyapunov Function:

$$V(z, \tilde{\theta}, \eta) = z_1^T z_1 + z_2^T z_2 + \tilde{\theta}^T \tilde{\theta} + \eta^T \eta$$

considering $\boldsymbol{u_{tot} = u_t + u_s}$ (where $u_s = u_{stable}$ is the controller that ensures stability) with $k_1 > 0$ and $k_2 > 0$ let's evaluate the derivative of V along the system trajectories:

$$\begin{aligned}
\dot{V} &= z_1^\top \dot{z}_1 + z_2^\top \dot{z}_2 - \tilde{\theta}^\top \dot{\hat{\theta}} + \eta^\top \dot{\eta} \\
&= z_1^\top(-k_1 z_1 + z_2) + z_2^\top \big(-k_2 z_2 + \tilde{\theta} + u_s\big) - \tilde{\theta}^\top z_2 + \eta^\top \left(S(p_u - p_{CoM})[u_t + u_s] + S(p_\theta - p_{CoM})\theta\right) \\
&= -z_1^\top k_1 z_1 - z_2^\top k_2 z_2 + z_1^\top z_2 + z_2^\top u_s + \eta^\top \left(S(p_u - p_{CoM})[u_t + u_s] + S(p_\theta - p_{CoM})\theta\right) \\
&\leq z_2^\top u_s + \eta^\top \left(S(p_u - p_{CoM})[u_T + u_s] + S(p_\theta - p_{CoM})\theta\right).
\end{aligned}$$

Now if we chose u_s such that:

$$\big(z_2^\top + \eta^\top S(p_u - p_{CoM})\big) u_s \leq -\eta^\top \left(S(p_u - p_{CoM}) u_t + S(p_\theta - p_{CoM})\theta\right) \tag{8}$$

Then we get that $\dot{V} \leq 0$, moreover (7) has solution thanks to the skew-symmetric nature of $S(\cdot; p_{CoM})$.

So in that way the closed loop dynamics is stable, so also is the z sub-dynamics, and we get both stability and bounded tracking error requirements.

Remarks:

- The term $\tilde{\theta}^T\dot{\tilde{\theta}} = \tilde{\theta}^T\dot{\hat{\theta}}$ because $\dot{\tilde{\theta}} = \dot{\theta} - \dot{\hat{\theta}}$ but the payload is constant so $\dot{\theta} = 0$.
- The control input u_t stabilizes the origin of the reduced space $(z, \tilde{\theta})$ (i.e. $z_1 \to 0$, $z_2 \to 0$, $\hat{\theta} \to \theta$). When we consider also η then the momentum dynamics $\dot{\eta}$ restricted to the set $\mathcal{S} = (z, \eta) : z = 0$ becomes an internal *zero dynamics*. This zero dynamics is unstable and we need the other part of the control u_s to stabilize it.
- The impact forces computed by (6) *may not guarantee feasible robot locomotion*. This is mainly due to having no considerations for the feasibility of the contact forces in the sense of the firction cone constraints.

7.2 Mass m different from 1

Note that we considered m=1 in this formulation, if this is not true we get:

$$\begin{aligned}
\boldsymbol{\dot{z}_1} &= \boldsymbol{-k_1 z_1 + z_2} \\
\boldsymbol{\dot{z}_2} &= \boldsymbol{-k_2 z_2 + \tilde{\theta}} \\
\boldsymbol{\dot{\hat{\theta}}} &= \boldsymbol{\frac{1}{m} z_2} \\
\boldsymbol{\dot{\eta}} &= \boldsymbol{S(p_u - p_{CoM})\big(m * u_t + u_s\big) + S(p_\theta - p_{CoM})\theta}
\end{aligned}$$

8 Stable Centroidal MPC for Robust Locomotion

In this section we provide an MPC algorithm to robustly solve our problem. In particular we use the control input u_t for adaptation(tracking) and gravity compensation, while we use optimization throught the MPC formulation to obtain the value of u_s enforcing stability constraint and satisfying additional constraints(as friction cone constraints for u_t).

MPC digital controller computes at each sampling time an optimal control sequence by solving a constrained optimization problem, in which the cost function involves the actual state given by some measurements (feedback).

8.1 MPC: Cost function

Let's consider the following cost functional:

$$\mathcal{J} = \sum_{k=0}^{n_p} T_{z_1(k)} + T_{\eta(k)} + T_{p_F(k)} + T_{u(k)} \tag{9}$$

where $n_p \geq 1$ is the *prediction horizon* of the MPC controller, $k \in \mathbb{Z}_{\geq 0}$ is the *time step* and we have:

$$T_{z_1(k)} = z_1(k)^T Q_1 z_1(k) \tag{10}$$

$$T_{\eta(k)} = \eta(k)^T Q_2 \eta(k) \tag{11}$$

for some positive definite matrices $Q_1, Q_2 > 0$ *penalizing linear and angular momentum errors.*

Additionally:

$$T_{p_F(k)} = (P_F(k) - P_F(k)^{des})^T Q_3 (P_F(k) - P_F(k)^{des}) \tag{12}$$

is a task penalizing the deviation of the feet contact locations p_F form a nominal contact location $p_F^{des}(k)$ at time k, assuming the presence of a high-level contact planner that generates only the contact location and timings (footstep trajectory planner).

Now we can better explain the meaning of the matrix Γ proposed in section 2:
Since the proposed controller assumes the knoledge of the contact sequence, it is possible to define the variable $\Gamma_i \in 0,1$ *for each contact.* Γ_i rappresents the contact state at a given istant. So $\Gamma_i(t) = 0$ indicates that the contact $i-th$ is not active at time t, while, when $\Gamma_i(t) = 1$ the contact is active. (*Note that we can use it also for considering the fact that a vertex of the foot is in contact, as we will do*).

We model the transition of a contact[5] from non-active to active using, for each contact location, a continuous variable subject to the following dynamics:

$$\dot{p}_{F_i} = (1-\Gamma_i)v_{F_i}$$

where v_{F_i} is the contact velocity. We can image that when the contact is active, i.e. $\Gamma_i = 1$, then the equation becomes $\dot{P}_{F_i} = 0$, so in other words, the contact location is constant if the contact is active.

Finally $\boldsymbol{T_{u(k)}}$ is a regularization task on the contact impact forces [5], in particular we want that the forces acting on the ***feet corners*** *are as symmetric as possible.*

As we said for a foot in contact with the ground, we can imagine that the exchanged forces are applied to the vertices of the rectangula foot. The net effect of these forces *is given by the sum of them* and we our goal is to have these forces as similar as possible.

So we use the task (13) that weights the difference of each contact force from the average, the expression of $T_{u(k)}$ is very long and tricky, so we proceed step by step:

Let's start considering *the force acting on vertex 1* v_1 *the foot 1*, we need to evaluate:

$$\Psi_{f_1,v_1} = \left\| \begin{bmatrix} \frac{1}{4}(f^x_{1,v_1} + f^x_{1,v_2} + f^x_{1,v_3} + f^x_{1,v_4}) - f^x_{1,v_1} \\ \frac{1}{4}(f^y_{1,v_1} + f^y_{1,v_2} + f^y_{1,v_3} + f^y_{1,v_4}) - f^y_{1,v_1} \\ \frac{1}{4}(f^z_{1,v_1} + f^z_{1,v_2} + f^z_{1,v_3} + f^z_{1,v_4}) - f^z_{1,v_1} \end{bmatrix} \right\|^2_{Q_4} = \left\| \begin{bmatrix} \frac{1}{4}\mathcal{F}^x_1 - f^x_{1,v_1} \\ \frac{1}{4}\mathcal{F}^y_1 - f^y_{1,v_1} \\ \frac{1}{4}\mathcal{F}^z_1 - f^z_{1,v_1} \end{bmatrix} \right\|^2_{Q_4} = \left\| \frac{1}{4}\mathcal{F}_1 - f_{1,v_1} \right\|^2_{Q_4}$$

We omitted the dependence from k, $\mathcal{F}$ is a short notation. In matricial form can be expressed as:

$$\Psi_{f_1,v_1} = \begin{bmatrix} \frac{1}{4}\mathcal{F}^x_1 - f^x_{1,v_1} & 0 & 0 \\ 0 & \frac{1}{4}\mathcal{F}^y_1 - f^y_{1,v_1} & 0 \\ 0 & 0 & \frac{1}{4}\mathcal{F}^z_1 - f^z_{1,v_1} \end{bmatrix} \begin{bmatrix} q_{41} & 0 & 0 \\ 0 & q_{42} & 0 \\ 0 & 0 & q_{43} \end{bmatrix} \begin{bmatrix} \frac{1}{4}\mathcal{F}^x_1 - f^x_{1,v_1} & 0 & 0 \\ 0 & \frac{1}{4}\mathcal{F}^y_1 - f^y_{1,v_1} & 0 \\ 0 & 0 & \frac{1}{4}\mathcal{F}^z_1 - f^z_{1,v_1} \end{bmatrix}$$

$$= \Phi^T_{f_1,v_1}\ Q_4\ \Phi_{f_1,v_1}$$

So following these reasoning the full term $\boldsymbol{T_{u(k)}}$ is given by:

$$\begin{aligned} T_{u(k)} = {} & \Phi^T_{f_1,v_1}\ Q_4\ \Phi_{f_1,v_1} + \Phi^T_{f_1,v_2}\ Q_5\ \Phi_{f_1,v_2} + \Phi^T_{f_1,v_3}\ Q_6\ \Phi_{f_1,v_3} + \Phi^T_{f_1,v_4}\ Q_7\ \Phi_{f_1,v_4} + \\ & + \Phi^T_{f_2,v_1}\ Q_8\ \Phi_{f_2,v_1} + \Phi^T_{f_2,v_2}\ Q_9\ \Phi_{f_2,v_2} + \Phi^T_{f_2,v_3}\ Q_{10}\ \Phi_{f_2,v_3} + \Phi^T_{f_2,v_4}\ Q_{11}\ \Phi_{f_2,v_4} \end{aligned} \tag{13}$$

8.2 MPC: Constraints

This cost function is associated, for all $k = 0, ..., n_p$, to the following constraints:

1. *The prediction model*: namely a Forward-Euler integration of the *unperturbed* dynamics:

$$\begin{aligned} p_{CoM}(k+1) &= P_{CoM}(k) + \Delta B h \\ h_\ell(k+1) &= h(k) + \Delta\left(\sum_{j=1}^{n_c} A_{u_j}(p_{u_j})\Gamma_j u_j(k) + m\overrightarrow{g}\right) \\ p_{u_i}(k+1) &= p_{u_i}(k) + \Delta([1-\Gamma_i]v_{u_i}) \end{aligned} \tag{14}$$

   Where Δ is the controller sampling rate, and p_{u_j}, v_{u_j}, Γ_j are the position, velocity and status of the contact at *the corner j* of the robot feet.

2. The coordinate change (5) ad the feedback relation over the prediction horizon:

$$\begin{aligned} z_1(k) &= p_{CoM}(k) - P^d_{CoM}(k) \\ z_2(k) &= k_1 z_1(k) + bh(k) - \dot{p}^d_{CoM}(k) \\ \eta(k) &= Ch(k) \\ u(k) &= u_t(k) + u_s(t) \end{aligned} \tag{15}$$

   where u_t(k) is the feedback (6) at time k and $u_s(t)$ is considered as a *decision variable* to find throught MPC.

3. *The stability constraints*: We do not have a mesure of the actual disturbance θ, so instead of enforcing the equality (7) we impose the following equivalent two constraints:

   (a) The first aims to stabilize the $(z, \tilde{\theta})$ sub-dynamics given the feedback $u = u_t + u_s$, so:

$$-z_1^T(k)k_1 z_(k) - z_2^T(k)k_2 z_2(k) + z_1^T(k)z_2(k) + z_2(k)^T u_s(k) < 0 \tag{16}$$

   This comes from the Lyapunov stability analisys such feedback.

   (b) The second complements the first by requiring the internal η dynamics to be stable:

$$||\eta(k+1)|| \leq ||\eta(k)|| \tag{17}$$

4. The contact forces feasibility constraints, *friction cone constraints*:

$$A\,{}^wR_F^T u(k) \leq b \tag{18}$$

   where ${}^wR_F$ (${}^wR_{F_c}$) is the rotation matrix associated with the impact forces w.r.t the world frame, and A, b constants depending on the friction coefficient. Note that this constraints must be applied to each of the four contact forces on the vertex of the feet[6]. We have:

$$A = \begin{bmatrix} 1 & 0 & -\bar{\mu} \\ -1 & 0 & -\bar{\mu} \\ 0 & 1 & -\bar{\mu} \\ 0 & -1 & -\bar{\mu} \end{bmatrix} \quad , \quad b = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad , \quad \bar{\mu} < \mu \quad \textit{To be verified}!!!!!$$

   since the contact force must satisfy $-\bar{\mu} f_z \leq f_x \leq \bar{\mu} f_z$ and $-\bar{\mu} f_z \leq f_y \leq \bar{\mu} f_z$.
   Remark: Note that the friction cone constraints are applied to the whole feedback $u = u_t + u_s$ and not only to u_s.

5. Constraint on the maximum allowable contact location adaptation error:

$$\ell_b \leq {}^{w}R_{F_c}((p_F) - (p_F)^d) \leq u_b \tag{19}$$

with ℓ_b , u_b being the lower and upper-bounds.

8.3 MPC remarks

The feedback $u = u_t + u_s$ with u_s being the solution to the MPC problem is able to solve our problem, *whenever yje optimal control problem is recursively feasible*.

We solve fot u_s treating u_t as a *feedforward control* while ensuring the friction cone constraints apply to their sum.
We can interpret this formulation as follows:

The MPC problem is the *"projection"* of the feedback u in a set defined by the force-feasibility and maximum contact adaptation errors constraints. For that reason the choice of the candidate Lyapunov function is relative to the unconstrained problem.
Moreover we can relax the problem and enforce constraints (15)-(16) only over the first predicted value and not necessarily over the whole horizon to enhance the computation time.

Note that we neglect the effects of the discretization on MPC problems because we assume to use a ZOH for control and measured signals fast enough that allow us to use the Lyapunov arguments in continuous-time. This is not restrictive in practice and is typically the case when dealing with robotics applications.

9 Python implementation

10 Simulation reuslts

11 References

[1] Stability Analysis and Design of Momentum-based Controllers for Humanoid Robots (Gabriele Nava, Francesco Romano, Francesco Nori and Daniele Pucci).
[2] S. Traversaro, D. Pucci, and F. Nori, "On the base frame choice in free-floating mechanical systems and its connection to centroidal dynamics," Submitted to Humanoid Robots (Humanoids), 2016, IEEE-RAS International Conference on, 2016. [Online].
Available: https://traversaro.github.io/preprints/changebase.pdf.
[3] D. Orin, A. Goswami, and S.-H. Lee, "Centroidal dynamics of a humanoid robot," Autonomous Robots,2013.
[4] I. Kanellakopoulos, P. V. Kokotovic, and A. S. Morse, "Systematic design of adaptive controllers for feedback linearizable systems," in 1991 American control conference. IEEE, 1991, pp. 649–654.
[5] 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," 2022 International Conference on Robotics and Automation (ICRA), Philadelphia, PA, USA, 2022, pp. 10412-10419
[6] Stéphane Caron, Quang-Cuong Pham, Yoshihiko Nakamura. Stability of surface contacts for humanoid robots: Closed-form formulae of the Contact Wrench Cone for rectangular support areas. ICRA: International Conference on Robotics and Automation, May 2015, Seattle, United States. pp.5107-5112, ff10.1109/ICRA.2015.7139910