Extension of IS-MPC to a LIP model with Virtual-Mass-Ellipsoid

Thesis Report 7

Under Supervision of:

Prof. Oriolo
Prof. Lanari
Dr. Sciancia
Filippo Smaldone

Omar Elsayed

May 9, 2021

1 Model

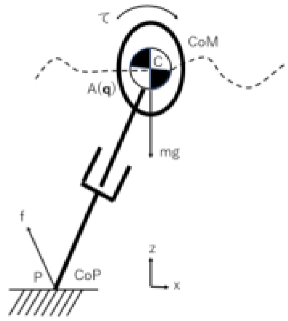

Figure 1: Virtual mass-ellipsoid model

Let r_c and r_p be the center of mass and ZMP in the world frame.

$$
\begin{aligned}
m\ddot{\boldsymbol{r}}_c =& f + m\boldsymbol{g} \\
(\boldsymbol{r}_z - \boldsymbol{r}_c) \times \boldsymbol{f} =& \dot{\boldsymbol{L}}_G \\
\dot{\boldsymbol{L}}_G =& \tau
\end{aligned}
\tag{1}
$$

The equations of motion are described as:

$$
\begin{aligned}
\ddot{x}_c &= \tfrac{g+\ddot{z}_c}{z_c - z_z}\left[x_c - x_z - \tfrac{\tau_y}{m(g+\ddot{z}_c)}\right] \\
\ddot{y}_c &= \tfrac{g+\ddot{z}_c}{z_c - z_z}\left[y_c - y_z + \tfrac{\tau_x}{m(g+\ddot{z}_c)}\right] \\
\ddot{z}_c &= \tfrac{g+\ddot{z}_c}{z_c - z_z}\left(z_c - z_z\right) - g
\end{aligned}
\tag{2}
$$

The equations can be separated into moment and non-moment inducing motions. This can be done by the following separation of equations. In addition, if we assume that the height of the center of mass is constant we obtain

$$
\begin{aligned}
\omega_0^2 &= \frac{g}{z_0} \\
x_c &= x_c' + \Delta x_c, y_c = y_c' + \Delta y_c, z_c = z_c' + \Delta z_c \\
\Delta x_z &= \frac{\tau_y}{mg}, \Delta y_z = -\frac{\tau_x}{mg}, \Delta z_c = 0
\end{aligned}
\tag{3}
$$

where z_0 represents the constant height of the center of mass of the humanoid. Substituting in the equations of motion 2, we obtain the equations of the separated dynamics of r_c' and Δr_c.

$$
\left\{
\begin{aligned}
\ddot{x}_c' &= \omega_0^2\left(x_c' - x_z\right) \\
\ddot{y}_c' &= \omega_0^2\left(y_c' - y_z\right)
\end{aligned}
\right.
\tag{4}
$$

$$
\left\{
\begin{aligned}
\Delta\ddot{x}_c &= \omega_0^2\left(\Delta x_c - \Delta x_z\right) \\
\Delta\ddot{y}_c &= \omega_0^2\left(\Delta y_c - \Delta y_z\right)
\end{aligned}
\right.
\tag{5}
$$

Equation 4 represents the non-moment inducing motion and equation 5 represents the moment inducing motion.

2 LIP

Plot 2 shows the actual trajectories of the mBase, mRobot and mTorso parts with respect to the desired Center of mass trajectory from the MPC block. Figures 3 and 4 show the error

$$\Delta X_c^k = X_{c,actual}^k - X_c'^k \tag{6}$$

$$\Delta Y_c^k = Y_{c,actual}^k - Y_c'^k \tag{7}$$

of the mBase and the mRobot body parts respectively with respect to world frame and foot frame.

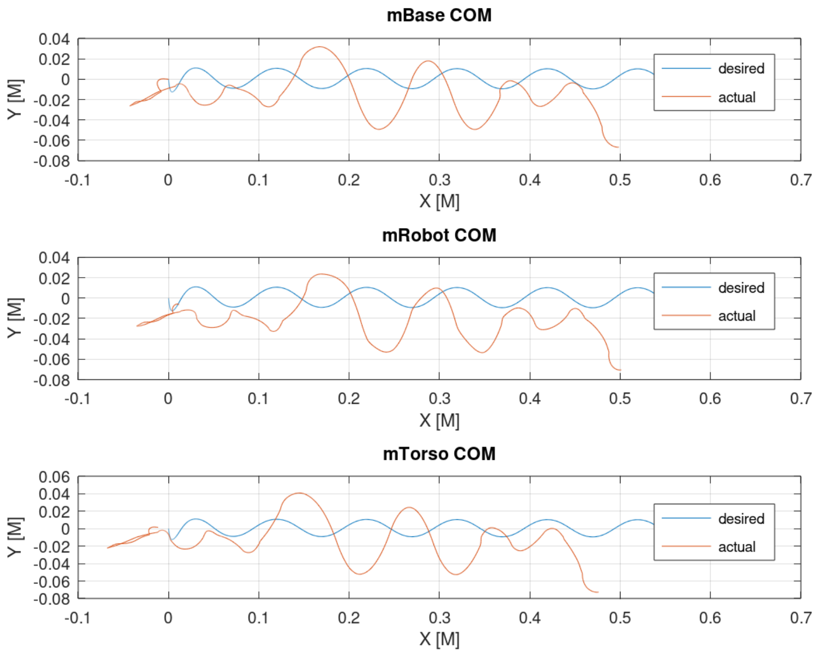

Figure 2: CoM trajectories of different body parts(base, robot and torso) with respect to desired trajectory

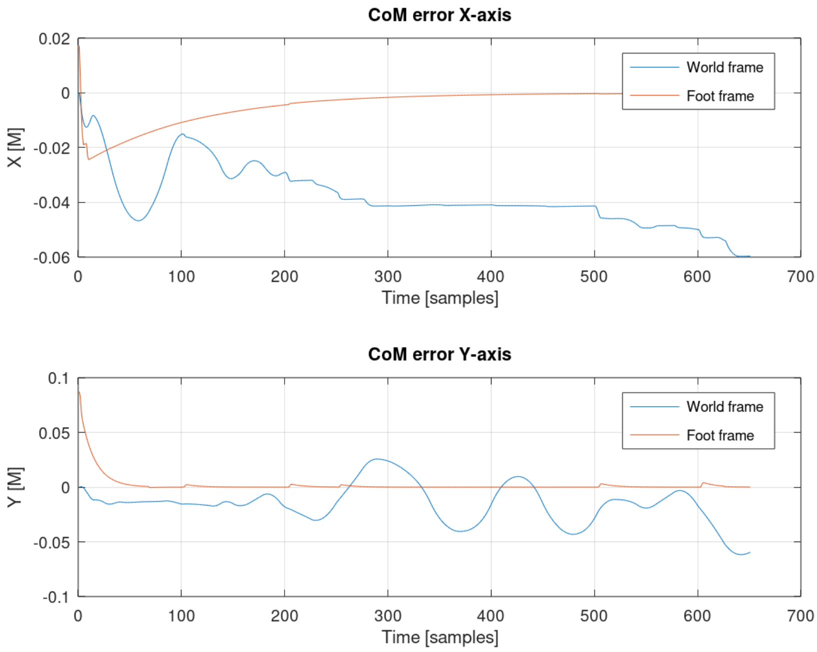

Figure 3: CoM error of mBase

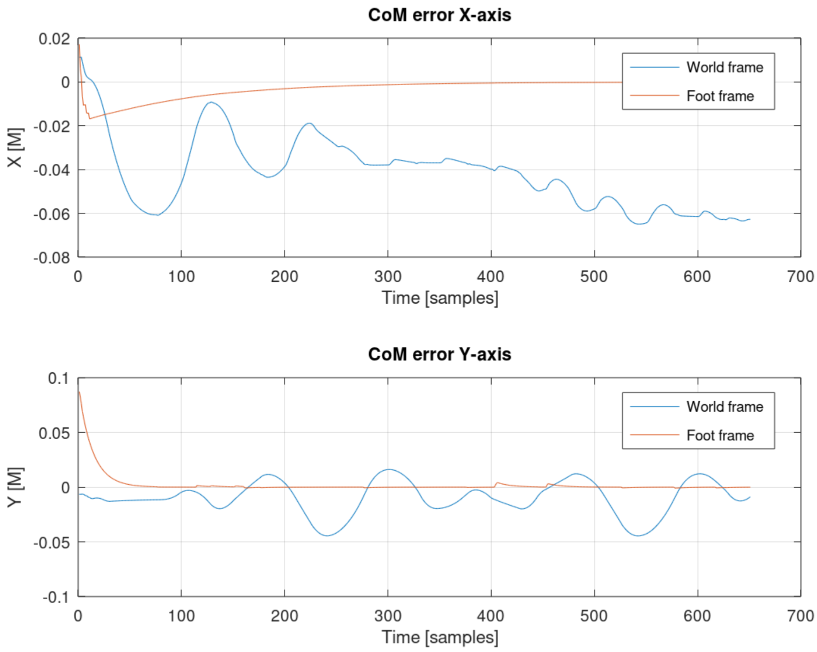

Figure 4: CoM error of mRobot

3 VIP

In order to actuate the Angular momentum pendulum, an error between an the actual measurement of the CoM and the desired CoM of the linear momentum LIP shown in equations 6 and 7. The following sections outline the results using different

$$\Delta X_c = k \cdot e, \qquad k > 0$$

where e is the error of the center of mass of mBase in the foot frame.

3.1 k = 1.0

Using $k\ = 1.0$, along with an angular constraint on the X and Y torso angles of $\pi/18$ is not feasible and the MPC will return a NaN when approaching the maximum allowed angles indicating that there is no solution to the MPC.

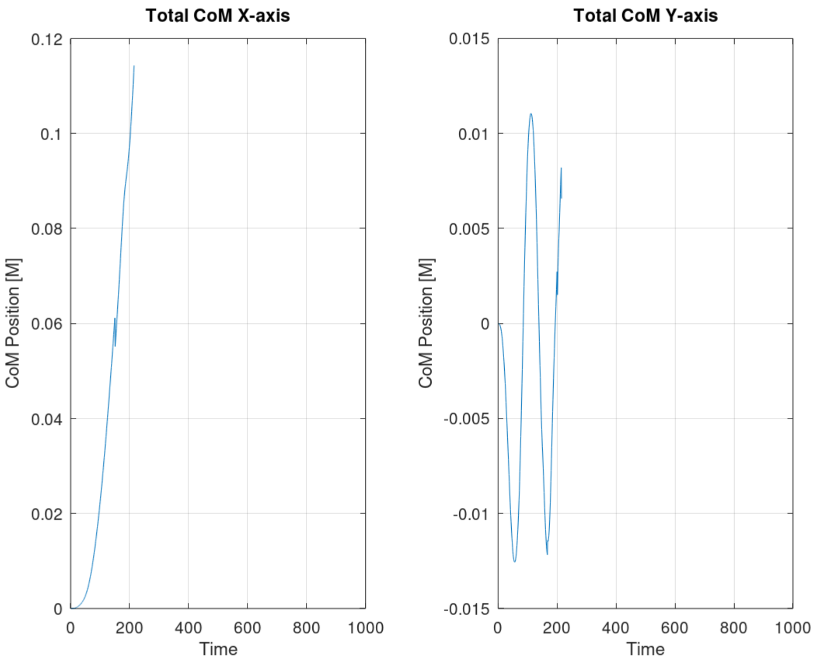

Figure 5: CoM trajectories

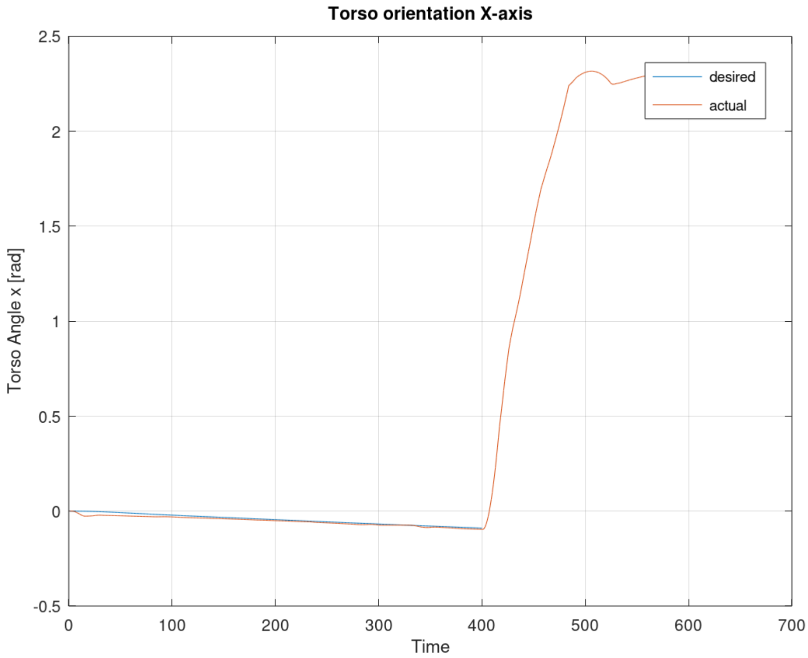

Figure 6: Torso Angle X-axis

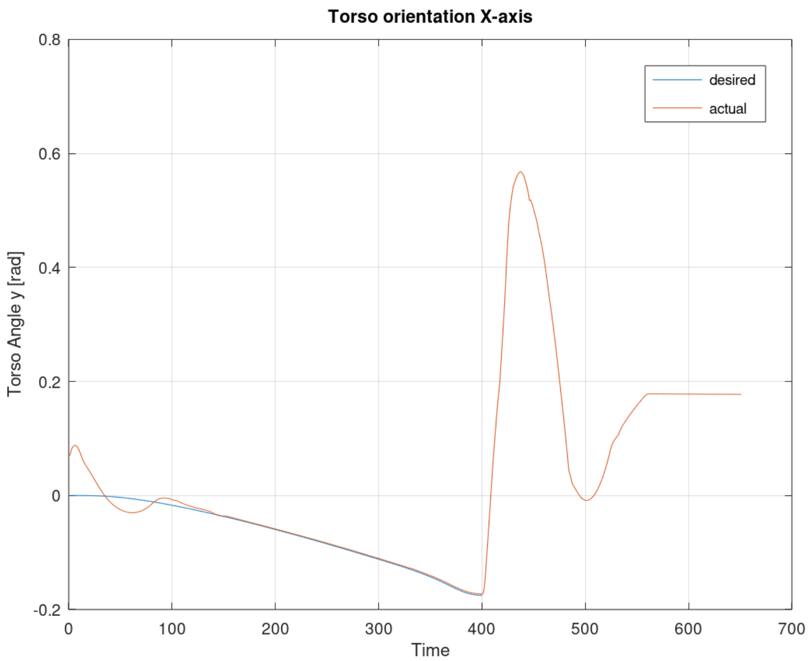

Figure 7: Torso Angle Y-axis

3.2 k = 0.01

Using a much lower gain seems better at first but it only takes the robot a few extra footsteps for the MPC to become infeasible when the orientation of the torso approaches the maximum (or minimum) allowed angle.

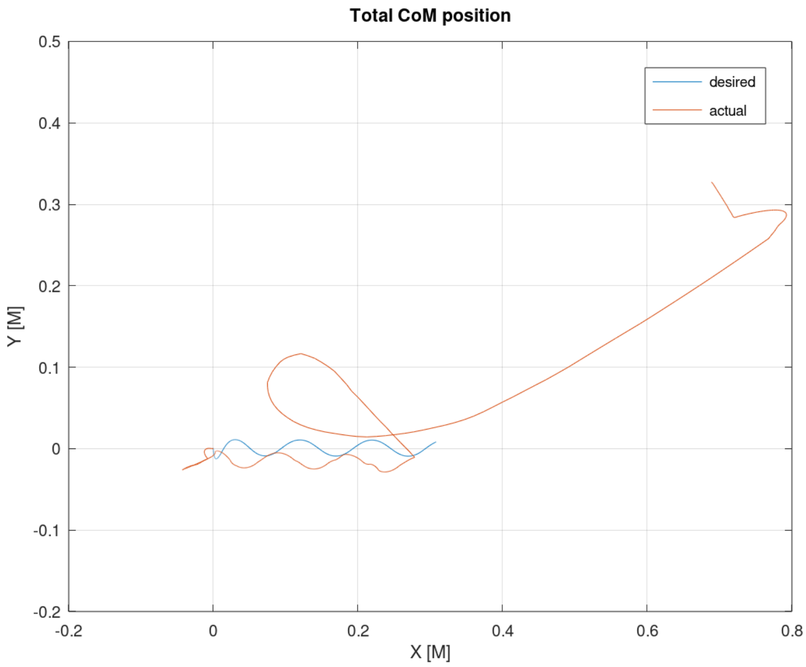

Figure 8: CoM trajectory

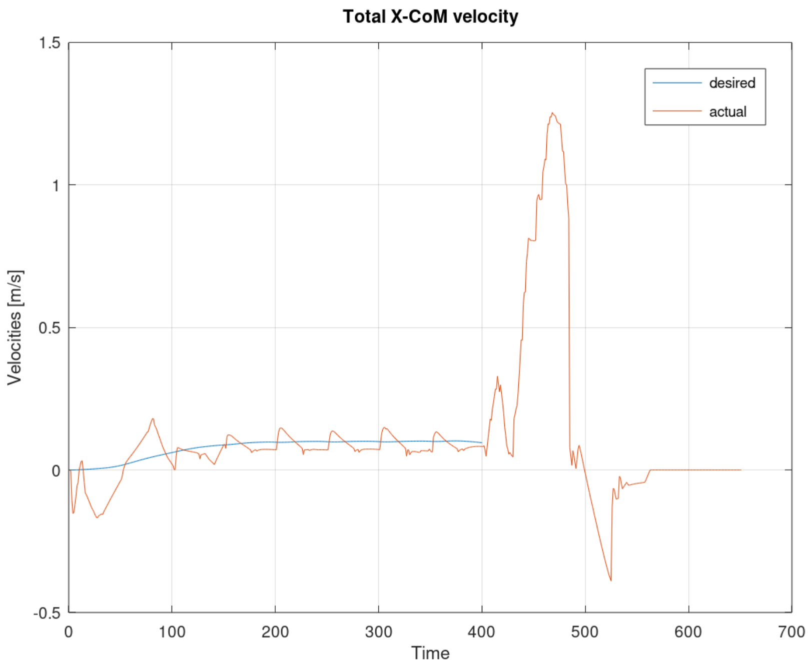

Figure 9: CoM Velocity in X-axis

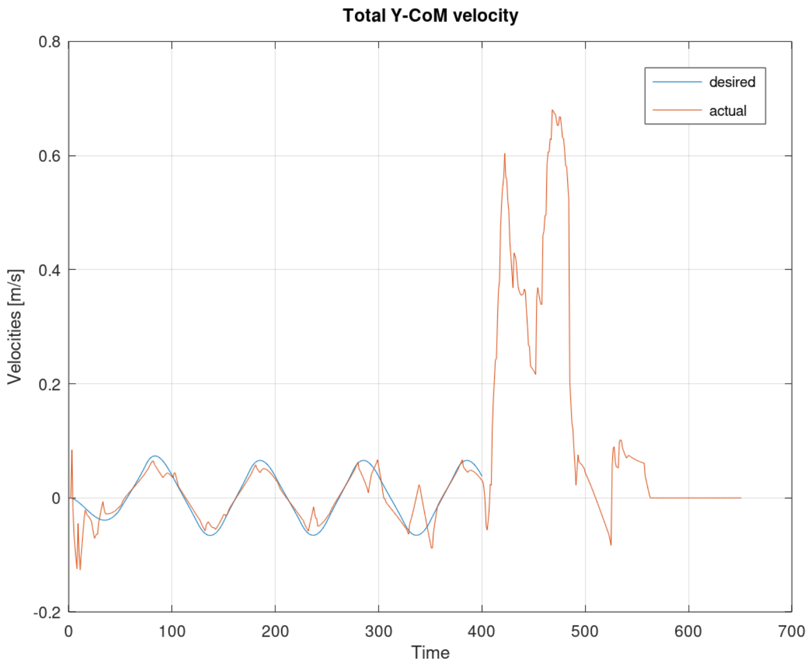

Figure 10: CoM Velocity in Y-axis

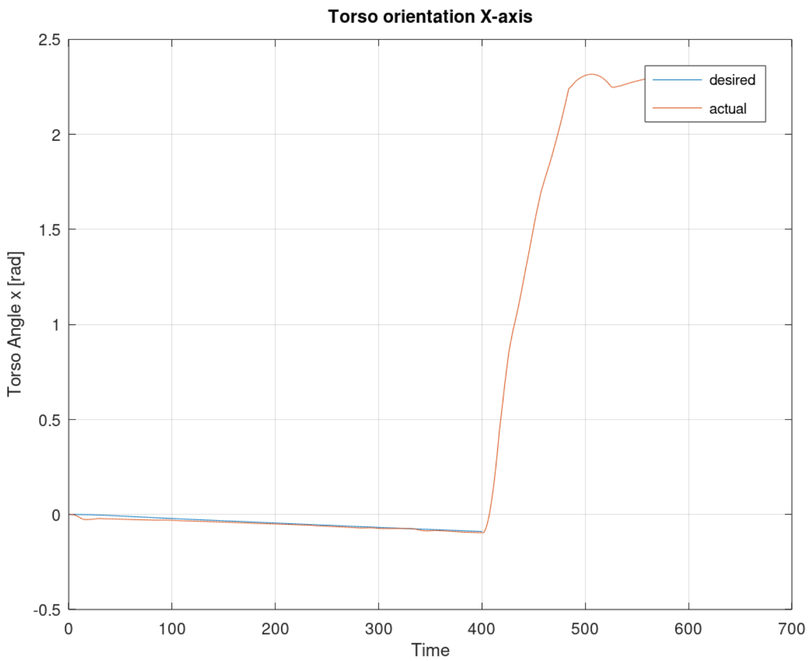

Figure 11: Torso Orientation in X-axis

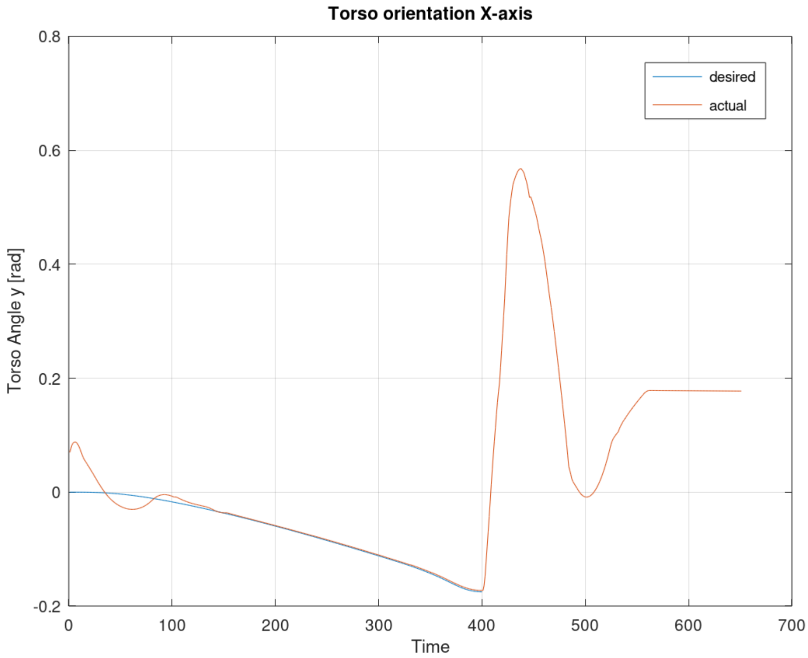

Figure 12: Torso Orientation in Y-axis

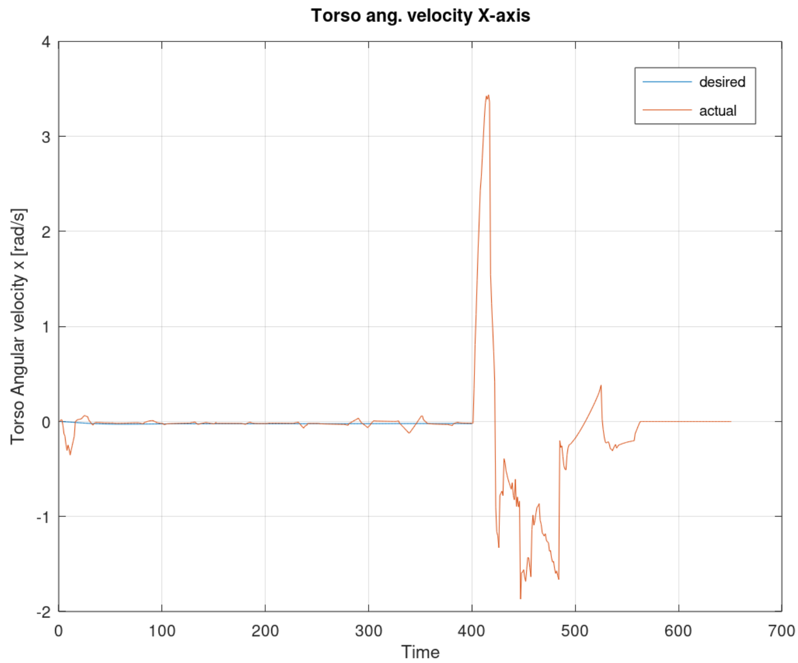

Figure 13: Torso Angular velocity around X-axis

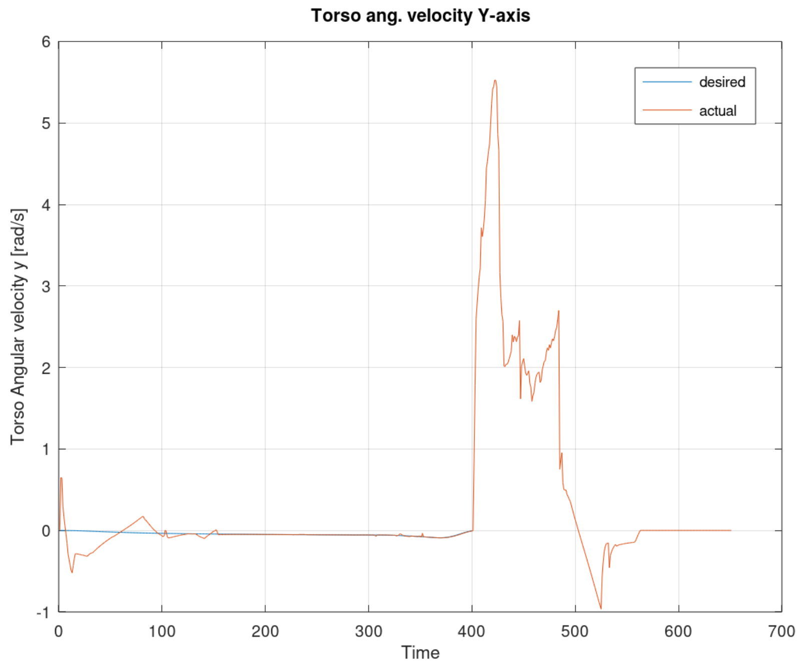

Figure 14: Torso Angular velocity around Y-axis

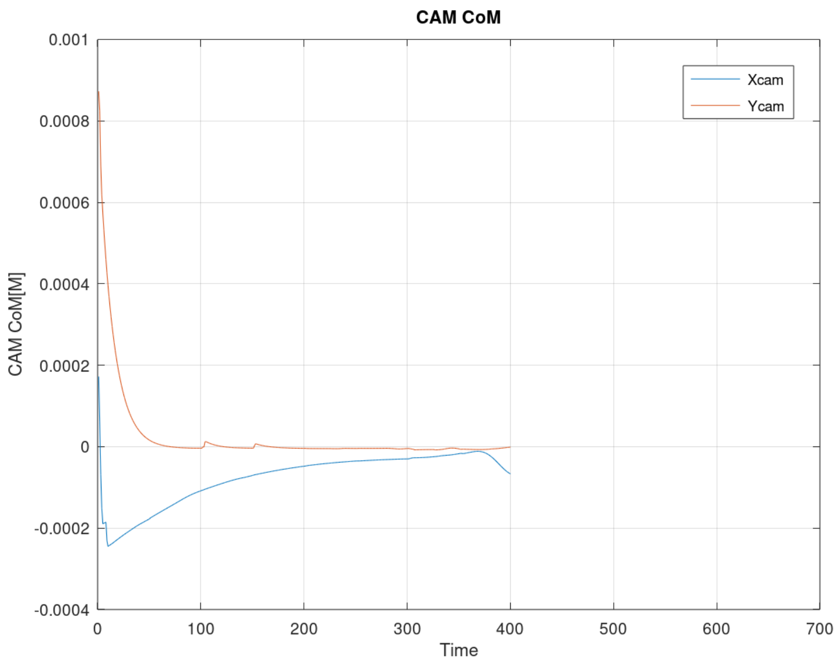

Figure 15: CoM of angular momentum LIP

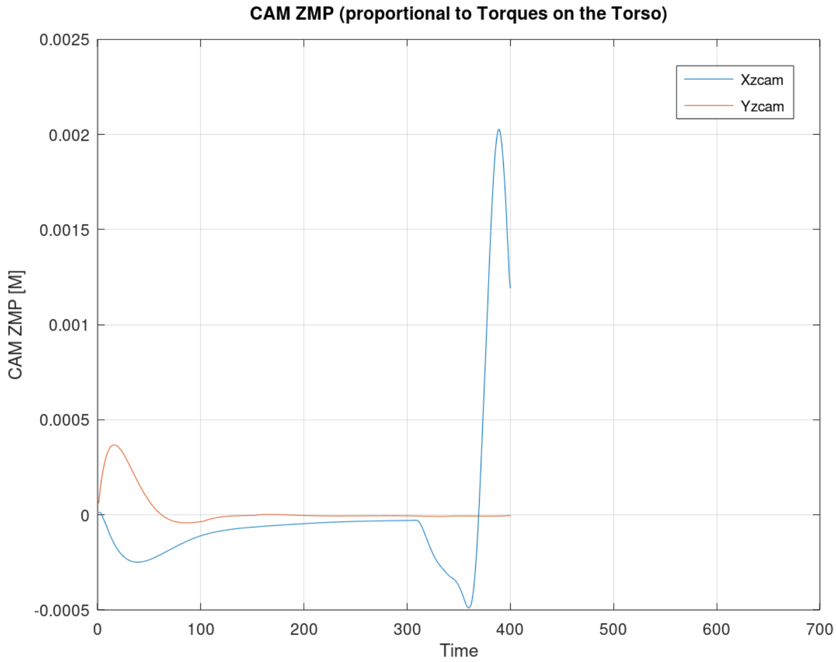

Figure 16: ZMP of angular momentum LIP

4 Comments

As shown before, the error input to the Angular momentum Pendulum will cause the MPC to be unsolvable, leading to the fall of the robot in the Dart simulations. Other approaches were made, for example running the simulation without the Angle constraint and activating the angular momentum from the second footstep (avoiding the first footstep where the robot stays in the same x-position), but the robot eventually becomes unstable and the desired trajectory unfeasible.