Bipedal Locomotion Control based on Divergent Component of Motion

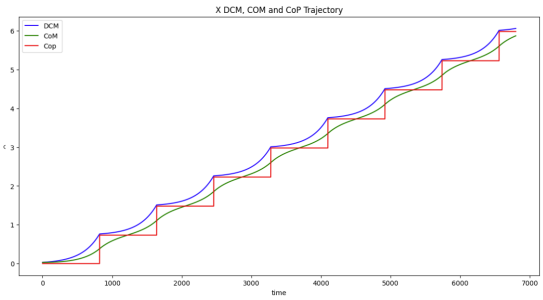
First Mini-Project of Legged Robots

Mischa Mez, Guillaume Spahr, Valmir Syla

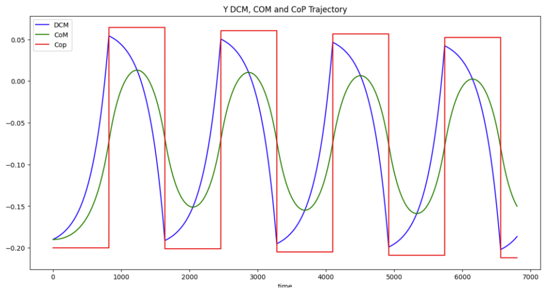
Robotics Master MA3
October 2024

1 Implementation of the Locomotion

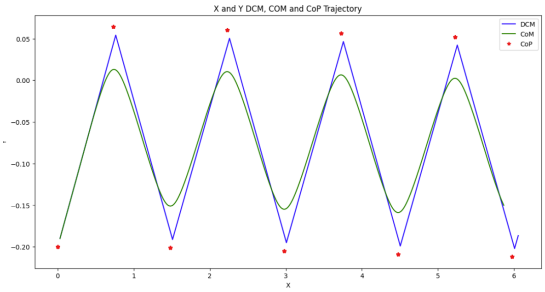
1.1 Observations of the toy example



((a)) DCM, CoM and CoP Trajectory of X component in function of time



((b)) DCM, CoM and CoP Trajectory of component in function of time



((c)) DCM, CoM and CoP Trajectory in the plane X-Y

Figure 1: Observations from the toy example simulation

In this simulation, we can see that the system is stable and that the CoM and the DCM stay between the two contact lines of the CoP.

1.2 Experiment

In this experiment, we apply an external force to the robot by throwing boxes onto it. The robot then attempts to regain stability by taking several steps. By analyzing the velocity and position of the Center of Mass (CoM), we can calculate the position of the Divergent Component of Motion (DCM) using the formula: $\xi = x + \frac{\dot{x}}{\omega}$. With these initial conditions, we input them into the optimization algorithm we previously completed, which computes the vector x. This vector determines the optimal position for the robot's foot placement, allowing it to stabilize by controlling the Center of Pressure (CoP).

2 Questions

2.1 Question 1

Based on equation (9), which physical parameters will affect the rate of divergence of the DCM dynamics?

We can recall equation (9):

$$\xi = (\xi_0 - \mathbf{cop}_0)\exp(\omega t) + \mathbf{cop}_0$$

The main parameter affecting the rate is the natural frequency ω, which depends on gravity and the height of the center of mass: $\omega = \sqrt{\frac{g}{\Delta z}}$. Increasing the height of the center of mass would make the DCM diverge slower, as it decreases the natural frequency. The rate of divergence also depends on the difference between the initial DCM and the initial center of pressure. The bigger the difference, the faster the divergence rate.

2.2 Question 2

The equation of motion for the Linear Inverted Pendulum (LIPM) gives the following solution for the position of the Center of Mass (CoM):

$$\dot{x} = A\exp(-\omega t) + B\exp(\omega t) + x_{\text{base}}$$

with

$$B = \frac{\dot{x}(0)/\omega + x(0) - x_{\text{base}}}{2}$$

where A represents the stable component, and B represents the unstable component of the motion. The Divergent Component of Motion (DCM) is defined as: $\xi = x + \frac{\dot{x}}{\omega}$. The DCM captures the unstable part of the CoM dynamics. To ensure stability, the foot placement is optimized such that the Center of Pressure helps counteract the growing $B\exp(\omega t)$ term ($COP = x_{\text{base}}$). By placing the foot near the DCM, the unstable component B is controlled (meaning it goes to 0), preventing it from growing exponentially, and the CoM converges to a stable trajectory over time. This is well described in the review document of the project that is on Moodle: "Indeed by setting the next foot step to ξ at the beginning of a step, time time evolution of x goes to zero"

2.3 Question 3

Even without an external push, errors can still arise due to several factors. As demonstrated in equation (9), the DCM includes a positive exponential term, indicating that even small initial differences between the DCM and the CoP will naturally grow over time. This inherent instability persists in the absence of external disturbances. Additionally, numerical integration errors can introduce small inaccuracies that accumulate over time and model imperfections, such as the simplification of system dynamics and the assumption of a constant CoM height, also contribute to these discrepancies.

2.4 Question 4

To recover balance, several strategies can be employed alongside the stepping strategy. For example, the robot's upper limbs can be used to shift its center of mass (CoM) forward, aiding in balance recovery. We however didn't code it so as to do that specifically. What we tryed however to tune and see the robot do is taking multiple smaller steps rather than a single large one when hit. This can help the robot stabilize itself more gradually and smoothly, reducing the intensity of the correction. Another approach that we saw working by tuning the variables is to lower the CoM by crouching after stepping, which increases stability and makes the robot less susceptible to gravitational forces. Finally, keeping the upper part of the body upright rather than allowing it to follow the motion can further enhance balance control.

2.5 Question 5

The DCM equation simplifies the control problem by isolating the unstable component, making the optimization process more computationally feasible and simple. For these reasons, the DCM model is well-suited for online motion planning and control.

In contrast, utilizing the full dynamic equation increases complexity, making optimization more challenging and complicating online motion planning.

2.6 Question 6

Adjust various parameters such as nominal parameters and the boundaries of the optimization variables, then report the maximum and minimum speeds at which the robot can achieve stable push recovery. During this process, you may need to modify the timing or magnitude of the external push to maintain stability. If any changes to the push are made, please provide detailed information on the push parameters, including its timing, magnitude, and direction.

We adjusted various parameters, such as nominal parameters and the boundaries of the optimization variables, to determine the maximum and minimum speeds for the x-axis at which the robot can achieve stable push recovery. During this process, we maintained the same push timing and magnitude to ensure consistent results. Below are the plots showing the center of mass (CoM) velocities in the x and y directions for each case:

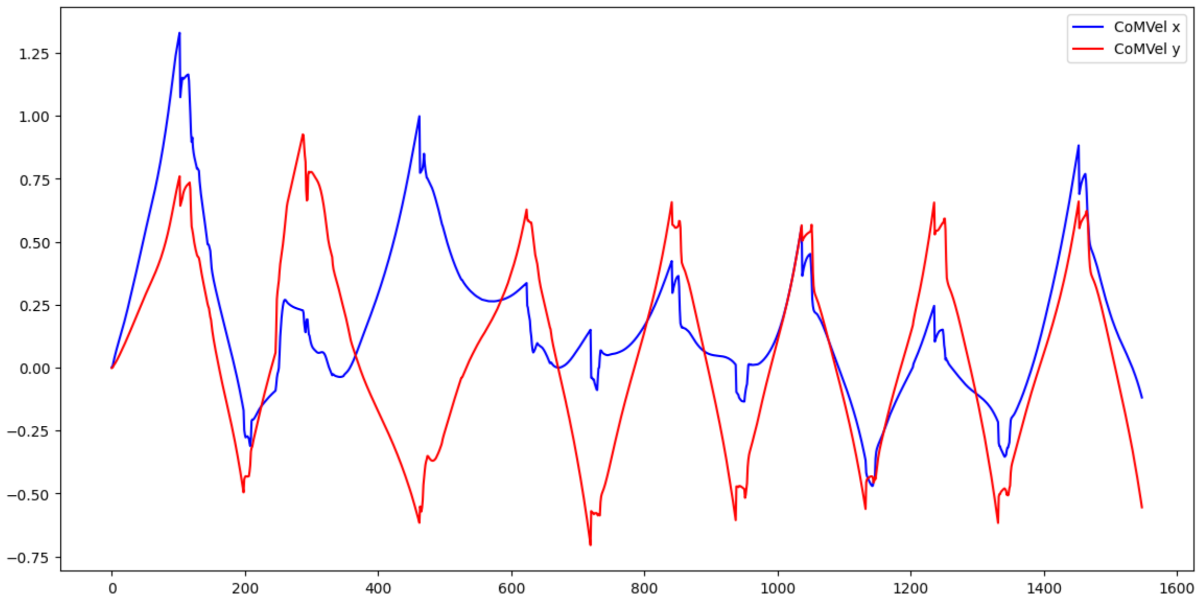


Figure 2: Velocities at each iteration for the minimum speed case

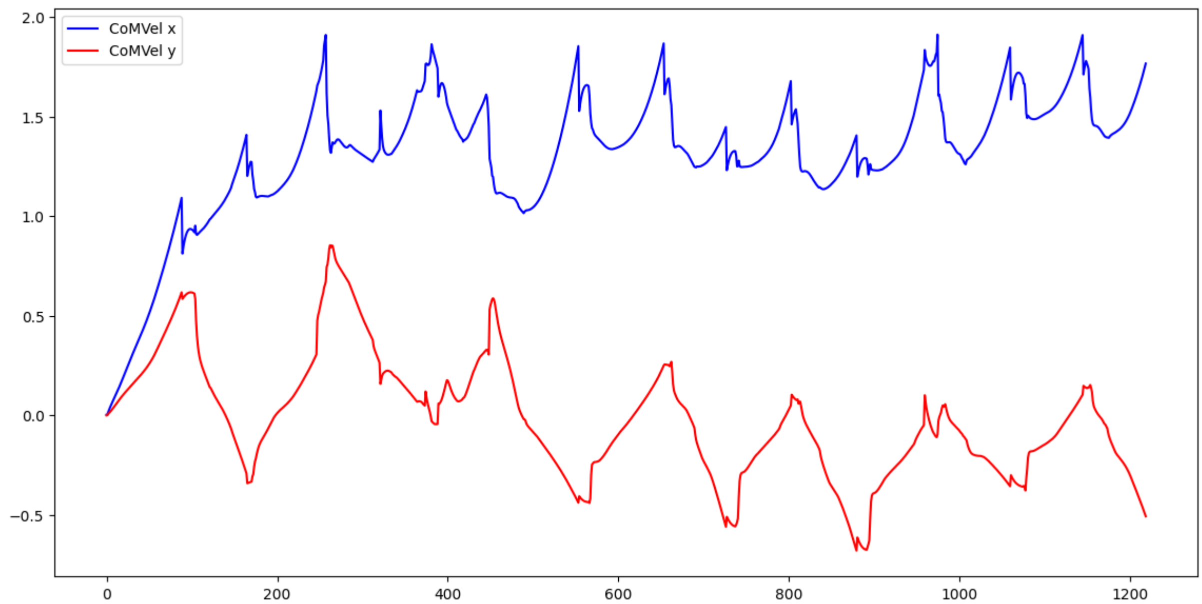


Figure 3: Velocities at each iteration for the maximum speed case

To obtain these results, we tested multiple parameter combinations. We observed that tuning these parameters can be time-consuming and that small changes can significantly affect the robot's walking behavior. This is expected, as we are exploring the limits of the robot's stability.

For the minimum velocity, the robot was almost stationary, neither advancing nor stepping back, after approximately 600 iterations. For the maximum velocity, the robot achieved a mean velocity of around 1.5 units in the x direction.

2.7 Question 7

To initiate a sequence of lateral recovery steps, the force magnitudes were adjusted as follows: force1 changed from [873,0,0] to [873,200,0], and force2 was modified from [0,1043,0] to [100,1043,0]. For the results, please refer to the video.

3 Videos

You can find the videos here:

`https://drive.google.com/drive/folders/1PZzzbv9mWq6soALNf_pfIa522Hezd8Z5?usp=sharing`