Final Project Report

Tarun Thathvik Paladugu
New York University- Tandon School of Engineering
New York City, United States of America
thathvik@nyu.edu

***Abstract*—In this project, a novel method incorporating Instantaneous Capture Point (ICP) and Model Predictive Control (MPC) were used to control a 2D-LIPM of a biped with velocity vector as the input.**

***Index Terms*—Instantaneous Capture Point, Model Predictive Control, Linear Inverted Pendulum Model**

I. Introduction

In the world of robotics, biped walking is one of the fields with the most interesting research. The absolute certainty that bipeds make intermittent contact with its environment has significant mathematical ramifications [8]. These complex dynamics of a biped can be studied and stabilized in parts by simplifying the dynamics and simulating this with heavy assumptions. In this project a similar implementation of a simplified model of a Linear Inverted Pendulum Model in 2D was simulated to walk in the x direction with no consideration of y and z movement.

To generate a stable motion for a biped with LIPM, one of the popular control laws is the Model Predictive Control (MPC) [2]. These controllers can be used with a cost function that aimed to minimum over the prediction or a constraint on the state of the system at the end of the horizon; sometimes a combination of both. In this model a combination of both is used, however several papers like [2][1][3] used only constraints on the state of the system at the end of the horizon. Since these the state of the system is completely dependent on the beginning and the end of the horizon, resulting in a motion that lacks the flexibility to perform secondary objectives such as absorbing perturbations or following speed requirements[2]. In the paper [2] this was still implemented with a slight adaptation.

However, these secondary tasks can be implemented by incorporating them directly into the model, which would require a more creative approach. In order for the robot to resist a fall, the instance at which the necessary control is to be applied needs to be identified. The fall needs to be detected, which requires a fall prediction. When the ICP moves from 0-step basin to 1-step basic, without a control action the robot will fall. This explained better in Section III. Stepping on the ICP will enable a safe stopping of the robot, and this concept is adapted into the model used. Since the desired foot position as an input, MPC can perform a remarkable job, a trajectory generated is implemented to address the desired velocity requirements.

With a Velocity input to generate trajectory online, and an adaptation of Instantaneous Capture Point to absorb perturbations, an MPC controller can perform the desired gait.

Structure: In the next section, a brief about the simulated model is given. Section III gives a detail of the implementation of both the adaptations. Section IV shows the results, and Section V, the conclusion.

II. Simulation model

To implement the biped walking, a 2D Linear Inverted Pendulum Model (LIPM) used in [5] was used **as is**. This model describes the dynamics of the Center of Mass (CoM) of the biped and assumes that the CoM moves on a plane parallel to the ground, with no angular moment. Since it is a 2D LIPM, the robot moves only in one direction, eliminating the dynamics of the biped in the y direction. Only the forward and backward motion of the LIPM is considered with x being the position of center of mass in the x-direction, and v being the velocity of the same. With that, eq(2) defines the forward motion of the CoM.

$$\dot{x} = v \tag{1}$$

$$\dot{v} = \omega(x - u) \tag{2}$$

In the eq(2), u as the Center of Pressure (CoP) position on the ground, and $\omega = \frac{g}{h}$, where g is the acceleration due to the gravitational force towards the ground, and h the height of the CoM from the ground. And the constructed control problem can be summarized in its canonical form as:

$$\min_{u_n} \sum_{n=0}^{N-1} x_n^T Q_n x_n + q_n^T x_n + u_n^T R_n u_n + r_n^T u_n \tag{3}$$

subject to,

$$x_{n+1} = Ax_n + Bu_n, \qquad x_0 = x(0) \tag{4}$$

and

$$G_n \begin{bmatrix} x_n \\ u_n \end{bmatrix} \leq h_n$$

where the dynamics is independent of n and the costs and bounds can change at every time step. These were the equations used in [5], and reused for this project without any change with the consent of Professor Ludovic Righetti.

In order to make the model trace the steps, a standard model predictive control with a horizon length 30 was used.

III. Implementation

Two different strategies were implemented to achieve a variable speed walking and push recovery, with the MPC as the main controller.

A. Push Recovery

For push resistance of this bot, a simple concept of Instantaneous Capture point was utilized. During the motion of a LIMP model, there exists a point on the ground, where placing the foot of the robot can stabilize the motion of the model [4]. This point is called Instantaneous Capture Point (ICP), or otherwise known as Extrapolated Center of Mass (XCoM) [7]. Since both the terms mean the same, they maybe used interchangeably. This point can be calculated using the eq(5).

$$x_{icp} = x + \frac{\dot{x}}{\omega_0} \tag{5}$$

where, x_{icp} is the x component of position of ICP on the ground, and $\omega_0 = \sqrt{\frac{g}{h}}$

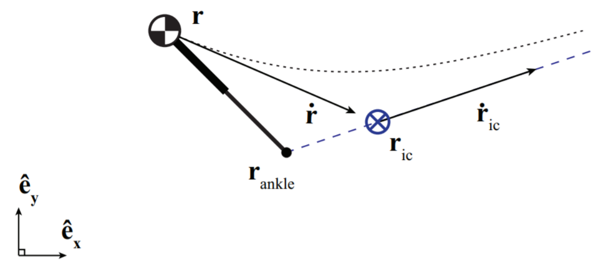

Fig. 1. Top view of the 3D-LIPM with point foot for a given initial state at time t[4].

1) Behavior of the ICP: Due to the regular dynamics of the bot, if the point is not captured, with time, the ICP moves in a straight line joining the CoP of the foot and the ICP in the direction away from the foot, as shown in the fig(1).

2) 0-step capturability: If the ICP falls inside the support polygon, then the robot can balance itself with the application of a good control like MPC; however, if the ICP falls outside the support polygon, the robot will have to take at least one step to protect itself from a fall. Since the model used here is a 2D model, we take that ICP to be inside the foot, as presented in the video submitted.

3) 1-Step Capturability: The model can come to stability within a step is the ICP falls within the steppable range. The steppable region is bounded by the maximum step length.

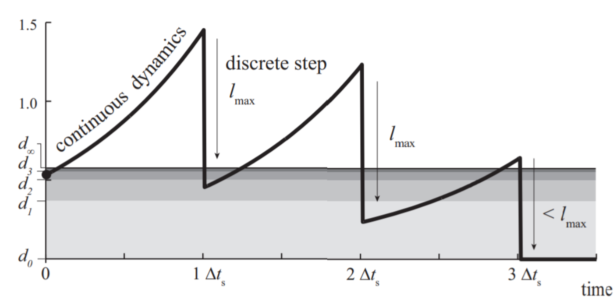

Fig. 2. X axis represents time, and the Y axis represents the difference between the foot position and the position of the ICP (Offset). Since this offset is greater than the maximum step length l_{max}, after the large step there still exists an offset, which behaves exactly as described in the behaviour, increasing in the same direction with time. In order to capture the controller takes multiple more steps to finally come to rest at zero offset

4) N-Step Capturability: When the ICP is outside the steppable region, the robot will need to take more than one step to stabilize itself from the push. Due to the behavior of the ICP the number of steps that are needed to be taken by the robot depends on the step maximum length and minimum step time, which are often defined based on the physical constraints of the robot. This is demonstrated in the fig(2), and if after N-steps the offset doesn't come to zero, then that point is non-capturable.

B. Variable Velocity

As mentioned in the section above (hyperlink maybe), the model is stabilized by the MPC around the given foot trajectory. However, this cannot be changed in the MPC directly without significantly modifying the considered dynamics of the model and the QP taken in the MPC. Since the goal is to implement a stable walking of the robot, with just the velocity at each moment as the input, a function that generates the trajectory of the footstep was implemented doing the same is defined. This function works with the principle of

$$v = \lambda f \tag{6}$$

where λ is the step length and f is the stepping frequency [6]. There are several conditions considered for better simulation.

1) Frequency: From testing, it was found that the MPC model could stabilize a considerably large range of velocity within a frequency of 1Hz. Therefore, λ is found using the 6 with f = 1Hz.

2) Step Length: Typically, this is based on the physical constraints on the model; however, due to the lack such constraints in this simulated model, we implement a maximum step length of 1.6 units. Due to this constraint, at velocities greater than 1.6 units/sec, will result in a higher frequency motion.

3) Step Time: For a more realistic simulation, a minimum step time of 0.5s is implemented, for this function. This constraint also directly effects the frequency of the stepping based on the speed.

The trajectory generated here is given as an input to the MPC which then stabilizes the model during the gait.

C. Adaptation

Due to movement of the robot, the position of ICP is always changing, and rarely stays inside the support polygon. Without any modification, the model would always predict falling and try to control it. To avoid this,the movement of the ICP during stable walking was carefully studied and a new constraint was adapted. The robot takes a step only when the ICP falls past the point of the next step, this would avoid the false prevention and enable the action only if the ICP goes beyond and is truly falling. Due to the faster step time (compared to the constraint in variable walking) allowed in the push recovery, the robot will be able to balance quickly with minimum steps; and continue moving at the trajectory velocity. The model is implemented in such a way that the last two seconds of the simulation, it stabilizes itself to stop.

IV. Simulation Results

The behavior of the MPC model used at a given foot position with a constant velocity is shown in the fig(3). Despite a slight noise in the position of CoM, the model is very stable with no unexpected stepping. This stable behavior was observed due to a horizon length of 30 in the MPC.

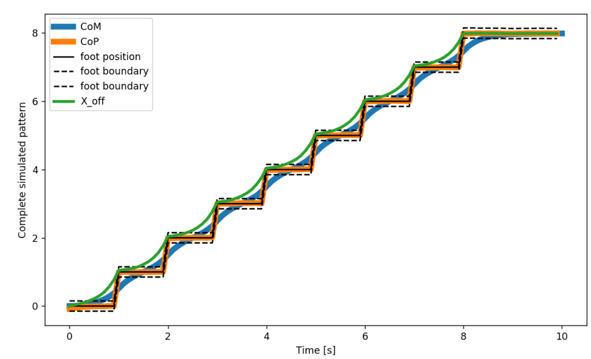

Fig. 3. Simulation with Constant Velocity

To observe the behavior of the system in the presence of a perturbation, a push is simulated by introducing a large disturbance to the position value of the CoM. It can be seen in fig(5) (top) that with a small perturbation, the model was able to stabilize with just one step, while fig(5) (bottom) shows numerous steps taken in the presence of a large perturbation. Fig(4) shows that the model responds well to the changing speed at the input. It can be observed from the fig(4)(top) that the mean velocity of the CoM is as desired. And fig(6) shows that the model can stabilize itself by stepping appropriately in presence of a perturbation during gait.

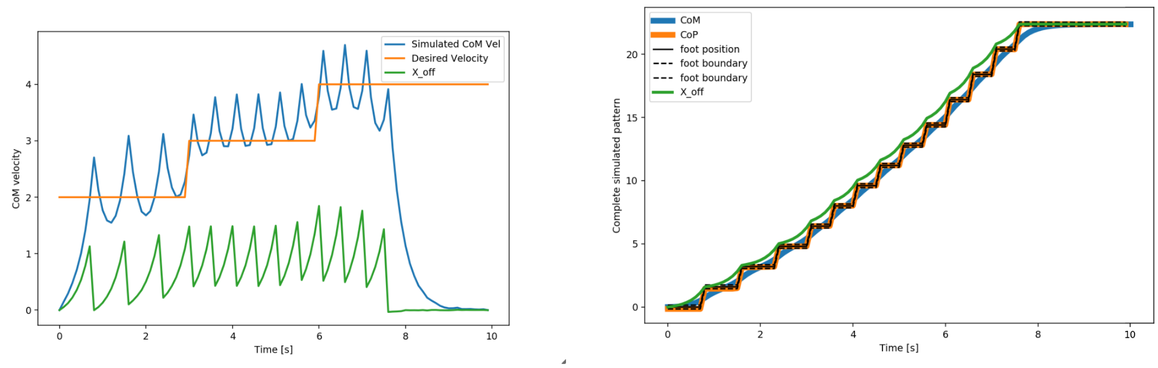

Fig. 4. (top) Shows the velocity of the CoM, ICP(XCoM), and the step like variable velocity input; (bottom) Shows the behaviour of the full model with this step like variable velocity input

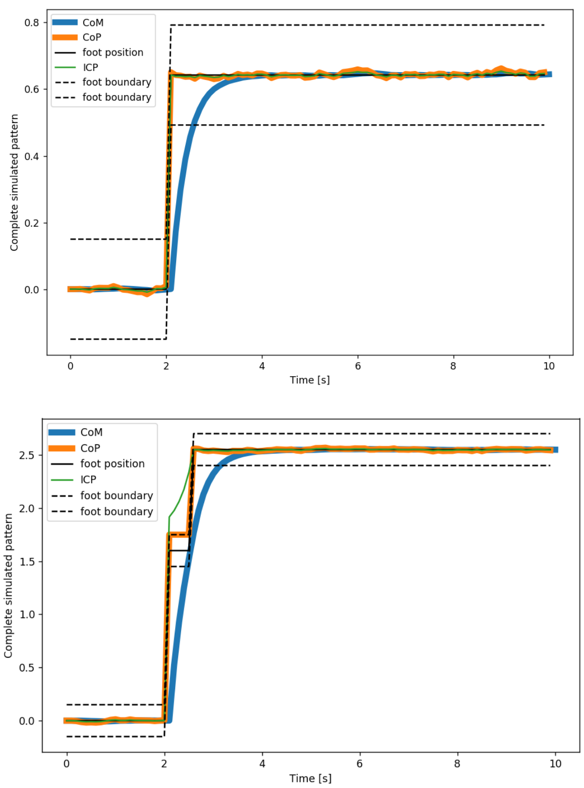

Fig. 5. (top) Small Perturbation; (bottom) Large Perturbation

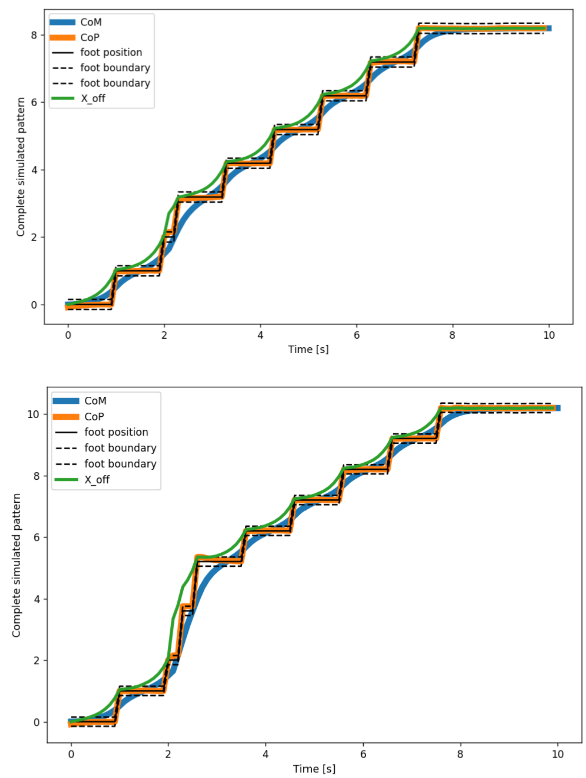

Fig. 6. (top) Small Perturbation; (bottom) Large Perturbation

It was observed that the robot cannot stabilize itself past a certain level of perturbation. However, this due to the various constraints applied, like the step length and the step time. Fig() shows that the model can be stabilized from any kind of perturbations just by placing its next step on the ICP, justifying the name “Instantaneous Capture Point”.

It was also observed that the maximum perturbations that the model could handle changed drastically with the time at which it was applied. If the perturbation was applied when the CoM becomes the slowest, then the resistance to perturbation is the highest. With no surprise, the maximum perturbation handled depends on the ICP position,

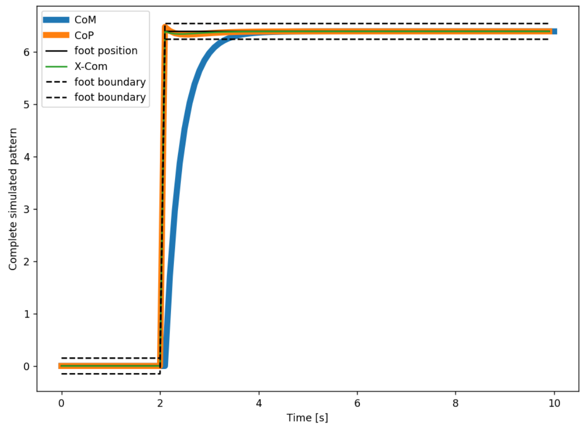

Fig. 7. By Stepping on the projected ICP, the model could stabilize even with such a large perturbation

becuase it is effected by the changing velocity and perturbation. If the perturbation is applied when the ICP is farthest from the foot, then the model can go only so much further, however, when the ICP is closest to the foot, the foot it can handle the largest perturbations. When the behavior of the model at a velocity that results in small step length was observed, it was resistant to large perturbations, confirming the same.

V. Conclusion

In this project, concept of ICP was implemented alongside a generic trajectory generator with an MPC to successfully perform a stable gait with only the input of velocity at a given instance, allowing a variable velocity, and a certain degree of resistance to perturbations. And in this model, the resistance to the perturbation depends on the offset of the ICP from the foot at the instant the perturbation is applied.

VI. Acknowledgement

Special thanks to Professor Ludovic Righetti, for giving me the opportunity to work on this project and providing the model with the MPC Collocation.

References

[1] H. Diedam et al. "Online walking gait generation with adaptive foot positioning through Linear Model Predictive control". English. In: Sep. 2008, pp. 1121–1126. ISBN: 2153-0866. DOI: 10.1109/IROS.2008.4651055. URL: https://ieeexplore.ieee.org/document/4651055.

[2] A. Herdt, N. Perrin, and Pierre-Brice Wieber. "Walking without thinking about it". English. In: IEEE, Oct 2010, pp. 190–195. ISBN: 2153-0858. DOI: 10.1109/IROS.2010.5654429. URL: https://ieeexplore.ieee.org/document/5654429.

[3] Andrei Herdt et al. "Online Walking Motion Generation with Automatic Footstep Placement". In: *Advanced Robotics* 24.5-6 (Jan. 2010), pp. 719–737. DOI: 10.1163/016918610X493552. URL: https://doi.org/10.1163/016918610X493552.

[4] Twan Koolen et al. "Capturability-based analysis and control of legged locomotion, Part 1: Theory and application to three simple gait models". en. In: *The International Journal of Robotics Research* 31.9 (Aug. 2012), pp. 1094–1113. DOI: 10.1177/0278364912452673. URL: https://doi.org/10.1177/0278364912452673.

[5] Righetti Ludovic. *Exercise 1 - Model Predictive Control*. 2020. URL: https://github.com/righetti/optlearningcontrol/blob/master/Spring2020/series3/Exercise%201%20-%20Model%20Predictive%20Control.ipynb.

[6] Hamed Razavi, Salman Faraji, and Auke Ijspeert. "From standing balance to walking: A single control structure for a continuum of gaits". en. In: *The International Journal of Robotics Research* 38.14 (Dec. 2019), pp. 1695–1716. DOI: 10.1177/0278364919875205. URL: https://doi.org/10.1177/0278364919875205.

[7] M. Vlutters, EHF van Asseldonk, and H. van der Kooij. "Center of mass velocity-based predictions in balance recovery following pelvis perturbations during human walking". English. In: *The Journal of Experimental Biology* 219.10 (2016), pp. 1514–1523. DOI: 10.1242/jeb.129338. URL: https://www.narcis.nl/publication/RecordID/oai:tudelft.nl:uuid:e8f97b35-beca-4f9d-85d9-9040a87cfe87.

[8] Pierre-Brice Wieber, Russ Tedrake, and Scott Kuindersma. "Modeling and Control of Legged Robots". In: ed. by Bruno Siciliano and Oussama Khatib. Springer Handbook of Robotics. ID: Wieber2016. Cham: Springer International Publishing, 2016. Chap. 48, pp. 1203–1234. ISBN: 978-3-319-32552-1.

DOI: 10.1007/978-3-319-32552-1_48. URL: https://doi.org/10.1007/978-3-319-32552-1_48.