

# Demonstrating a Robust Walking Algorithm for Underactuated Bipedal Robots in Non-flat, Non-stationary Environments

Oluwami Dosunmu-Ogunbi, Aayushi Shrivastava, Jessy W Grizzle

**Abstract**— This work explores an innovative algorithm designed to enhance the mobility of underactuated bipedal robots across challenging terrains, especially when navigating through spaces with constrained opportunities for foot support, like steps or stairs. By combining ankle torque with a refined angular momentum-based linear inverted pendulum model (ALIP), our method allows variability in the robot’s center of mass height. We employ a dual-strategy controller that merges virtual constraints for precise motion regulation across essential degrees of freedom with an ALIP-centric model predictive control (MPC) framework, aimed at enforcing gait stability. The effectiveness of our feedback design is demonstrated through its application on the Cassie bipedal robot, which features 20 degrees of freedom. Key to our implementation is the development of tailored nominal trajectories and an optimized MPC that reduces the execution time to under 500 microseconds—and, hence, is compatible with Cassie’s controller update frequency. This paper not only showcases the successful hardware deployment but also demonstrates a new capability, a bipedal robot using a moving walkway.

## I. INTRODUCTION

Bipedal robots exhibit substantial potential for aiding humans across various sectors, including hospitals, disaster management, factories, and construction sites. However, these environments often present challenges with constrained footholds, impeding conventional foot placement techniques due to limited space available for robot stepping.

Our approach introduces a dynamic variation that enables the robot’s center of mass height to adapt within each step, addressing the limitations posed by confined footholds. We first introduce our methodology and present simulation results in the SimMechanics environment in [1]. While these preliminary results are commendable, a significant challenge lies in transitioning from simulation to real-world applications. Bridging this gap is hindered by persistent difficulties in accounting for factors such as motor friction, sensor inaccuracies, and unforeseen variables that are not considered in simulation environments. Notably, deploying controllers on Cassie hardware poses a critical challenge due to the stringent maximum controller execution time of 0.5 milliseconds. The nominal model predictive controller (MPC) responsible for computing ankle torque demands an execution time exceeding this limit. In this paper, we discuss how we overcome these challenges and successfully implement our novel control method on the Cassie bipedal robot hardware.

### A. Background

Overly specific models can be computationally burdensome and lack versatility across robotic systems. Simplified models enable diverse control strategies, but may not

capture the primary dynamics effectively, limiting system agility. Bridging the sim-to-real gap remains challenging. We build on the Angular Momentum Linear Inverted Pendulum (ALIP) model from [2], [3], enhancing it to better accommodate changes in center of mass (CoM) height, as detailed in [1].

Several scholars, including Fu et al. [4] and Caron et al. [5], have explored the development of controllers tailored for fully actuated humanoid robots with 32 and 34 Degrees of Freedom (DoF) respectively, focusing on stair navigation. In [6], the authors devised open-loop stair gaits for the 3D underactuated 20 DoF Cassie bipedal robot, omitting investigation into closed-loop control. In [7], researchers managed to simulate human-like movements during planned and unplanned descents on the Cassie bipedal robot. Our paper aims to enhance the navigational capabilities of the Cassie biped by achieving an asymptotically stable periodic gait on non-flat, non-stationary terrain.

Previous research by Siekmann et al. [8] employed reinforcement learning to devise a closed-loop controller for the Cassie bipedal robot, treating changes in terrain elevation as an unpredictable perturbation. However, this approach resulted in significant impacts, posing potential risks to the robot’s structural integrity. The authors of [9] also explore dynamic walking on constrained footholds on level ground using control barrier functions. Dai et al. [10] tackled similar challenges by developing a dynamic walking controller for non-flat constrained footholds, including staircases, by regulating the underactuated robot’s vertical Center of Mass (CoM). We implement an alternative approach to navigation in varied terrain, utilizing the often-overlooked stance ankle motor alongside virtual constraint-based control.

The authors of [11] integrate the ankle torque model in their application of a terrain-aware controller also capable of walking on constrained footholds with successful implementation on hardware. In [12], the authors developed a novel controller to enable navigation over rough and challenging terrains for the actuated Spring Loaded Inverted Pendulum Model. This paper seeks to prove the validity of our novel robust controller on hardware to extend on initial SimMechanics results.

While the utilization of virtual constraints in legged locomotion is not novel [13], [14], [15], [16], [17], [18], [19], applying them to a non-flat environment for bipedal robot navigation, as we do in this work, presents an innovative application.

## B. Contributions

- 1) Presentation of the nominal trajectories tailored for various inclined planes.
- 2) Implementation of a foot placement strategy for lateral stabilization of the robot that can be implemented in real-time.
- 3) Enhancement of the Model Predictive Controller's efficiency in computing ankle torque for the sagittal plane. This involves linearizing impacts around nominal trajectories and the strategic offloading of computations to a secondary computer and the utilization of UDP communication between the secondary computer and Cassie.
- 4) Validation of the full controller on the Cassie bipedal robot hardware.

## II. REVIEW OF THE CONTROL PHILOSOPHY

In this section, we review the control philosophy outlined in our previous work [1] and elucidate the enhancements made to ensure its effectiveness in conducting hardware experiments.

The Cassie biped has 20 DoF to control. During single support, where one foot is grounded and the other is free, nine DoF are subject to holonomic constraints: four from Cassie's springs (two per leg) and five from the stance foot. Thus, we are left with 11 DoF to control and ten actuators. The robot is therefore underactuated.

We employ Passivity-Based Control (PBC) to regulate an additional nine DoF, utilizing nine of Cassie's actuators to track nominal trajectories generated offline. The tenth actuator is assigned to regulate one DoF, specifically the stance toe actuator, through a model predictive controller (MPC). The last DoF is managed via a lateral foot placement approach.

To facilitate the successful execution of the hardware experiments described in this paper, several critical adjustments were made. Firstly, new nominal trajectories were generated to align with the new experimental scenarios discussed in this study, as discussed in detail in Section III. Subsequently, enhancements were strategically implemented to enhance the efficiency and precision of the MPC algorithm. This involved transferring its computations to a secondary computer, integrating a real-time nonlinear optimization tool, and devising a new impact map based on linearization around the nominal trajectories, as elucidated in Section V. Finally, a real-time method was devised to compute the lateral foot placement strategy, detailed in Section IV.

## III. NOMINAL PERIODIC ORBIT

We utilized Fast Robot Optimization and Simulation Toolkit (FROST) [20] to generate nominal trajectories for Cassie, leveraging its advanced framework designed for optimizing and simulating the full-body dynamics of bipedal walking robots. FROST seamlessly integrates virtual constraints-based feedback controllers and employs a Wolfram Mathematica backend to symbolically generate expressions for multi-domain system dynamics and kinematics.

These symbolic expressions are then translated into C/C++ code, compiled into \*.MEX files under MATLAB, enhancing computational speed critical for hardware implementation.

FROST conceptualizes dynamic bipedal walking as a hybrid system, combining both continuous phases and discrete transitions, represented by a Directed Graph. A notable feature is its use of state-of-the-art direct collocation approaches for gait optimization, ensuring swift and dependable convergence. The default control law of FROST relies on virtual constraints-based feedback controllers. This toolkit emerges as a versatile and efficient resource for researchers exploring bipedal robot control and dynamics optimization.

In our study, FROST was employed to generate multiple nominal trajectories for Cassie. We generated trajectories for various scenarios, including marching in place, walking forward, and transitioning from walking on flat ground to different inclines (4 degrees, 8 degrees, 15 degrees, and 20 degrees). Utilizing Bézier curves [21], we approximated optimized trajectories for the nine virtual constraints (defined in [1]), as well as the angular momentum and CoM angle crucial for MPC computation for ankle torque, as detailed in [1].

### A. Generating Nominal Trajectories with FROST

Figure 1 illustrates animation outtakes from a nominal trajectory for non-flat terrain designed using FROST and a full-order model of the Cassie biped. The trajectory encompasses two steps, accommodating both right and left foot stance phases. The generation of this trajectory involved addressing several key constraints used for all the nominal trajectories.

A periodicity constraint was imposed at the conclusion of the second step, enabling the use of this trajectory to traverse an unlimited number of steps through the simple repetition of the nominal trajectory. Another constraint was incorporated to minimize torque on the stance ankle motor. This strategic addition provides increased adaptability when applying the trajectory to an actual Cassie robot. Our hypothesis posited that a nominal trajectory minimizing stance ankle torque would translate to minimal ankle torque requirements for the physical robot when walking unperturbed, allowing more room to overcome disturbances by utilizing stance ankle torque output. This hypothesis was validated, as discussed in Section VI.

In Figure 2, we present plots depicting the angular momentum and CoM angle, comparing the results computed by FROST for the nominal trajectories against the values obtained by the new variation of the ALIP model. As anticipated, the values are similar. This shows that the trajectories generated by FROST satisfy the ALIP model.

### B. Bézier Curves

We translate the numerical values of the nominal trajectories generated by FROST and express them in the language of Bézier curves so that they can be more readily used and manipulated by our controller. A Bézier curve is a mathematical representation of a curve defined by two or

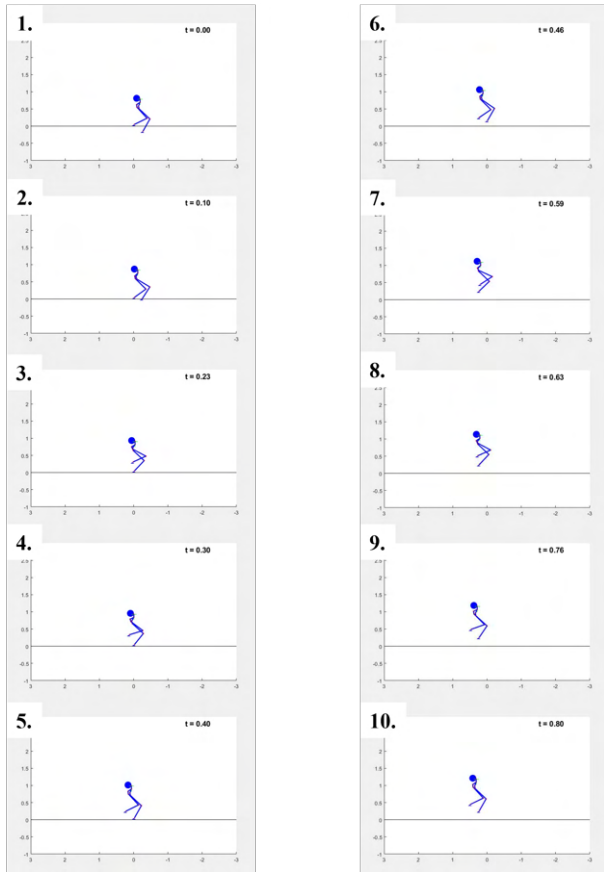


Fig. 1. Outtakes of a nominal trajectory for a non-flat terrain generated by FROST using a full-order model of the Cassie biped. The FROST animation is restricted to only showing a flat line at  $y=0$ . However, one can see that a non-flat trajectory is being followed by observing how the feet are placed with respect to the  $y=0$  line.

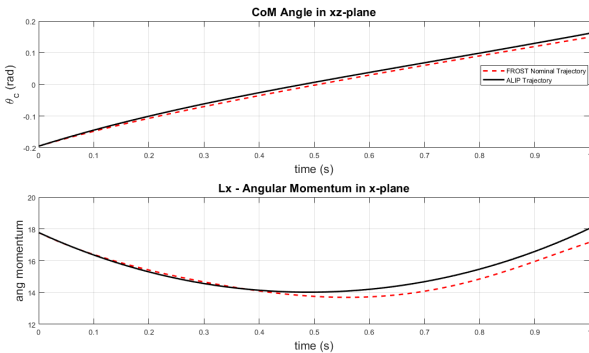


Fig. 2. Comparison of the nominal trajectory generated by FROST and ALIP model of angular momentum and CoM angle.

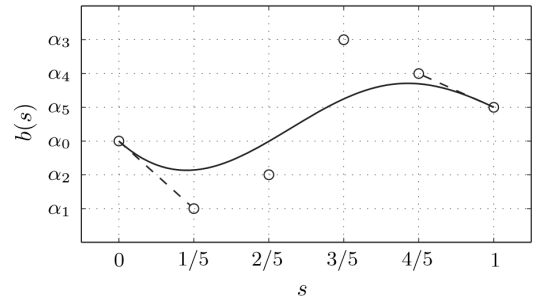


Fig. 3. Graph of a Bézier Curve of Order 5 [22].

more “control points,” which may be situated either on the curve itself or externally [22].

Figure 3 illustrates a Bézier curve of order 5, featuring six control points (the order plus one). The curve lies within the convex hull of these control points. Notably, the curve commences at  $b(0) = \alpha_0$  and concludes at  $b(1) = \alpha_5$ . This alignment is intentional, as for all Bézier curves,  $b_i(0) = \alpha_0^i$  and  $b_i(1) = \alpha_M^i$ , signifying that the  $i^{\text{th}}$  Bézier curve starts at the first coefficient  $\alpha_0^i$  and concludes at the last coefficient  $\alpha_M^i$ .

By incorporating Bézier curves into our controller for representing the nominal trajectories from FROST, we gain the ability to finely manipulate these trajectories through adjustments to their control points. For instance, to extend the time the swing foot takes to ascend and avoid contact with an obstacle (e.g., a stair), we strategically reposition the interior control points associated with the sagittal plane motion of the swing foot, moving them closer to the initial control point. This adjustment effectively prolongs the swing foot’s retention of its sagittal position as it elevates in the  $z$ -plane. The utilization of Bézier curves empowers us to make nuanced refinements to the nominal trajectories, such as this, without the need for the time-intensive task of regenerating trajectories through offline optimization.

#### IV. LATERAL STABILIZATION OF THE ROBOT

We stabilized the lateral motion of the Cassie biped by using the angular momentum-based foot placement strategy developed in [3], but with the new ALIP model derived in our work [1]. This section derives the foot placement strategy for the new ALIP model.

Unlike the previous version of the ALIP model, the new ALIP model does not have a closed-form solution. Thus, a numerical approach must be implemented for a foot placement strategy. To ensure that this approach can be implemented on hardware, we developed a look-up table to be used in real-time applications. This look-up table approach renders the execution time for the lateral foot placement controller to be less than  $5 \mu\text{s}$ , critical to ensure that we adhere to Cassie’s stringent 2 kHz control-update frequency.

The strategy to compute lateral foot placement numerically is outlined as follows:

- 1) Use Euler method for numerical integration to estimate the angular momentum at the end of the **next** step, **before** impact, for two lateral foot positions  $y_{st \rightarrow sw}^1$  and  $y_{st \rightarrow sw}^2$ , where  $y_{st \rightarrow sw}^1$  and  $y_{st \rightarrow sw}^2$  are close in value. We will call these two angular momentum values  $L_1$  and  $L_2$ . We pick the first lateral position  $y_{st \rightarrow sw}^1$  based off of the nominal trajectory and the second lateral position  $y_{st \rightarrow sw}^2$  to be a value close to the first.
- 2) Define a desired angular momentum  $L_{des}$  that we assume to be approximately along the line between Point 1 ( $y_{st \rightarrow sw}^1, L_1$ ) and Point 2 ( $y_{st \rightarrow sw}^2, L_2$ ).
- 3) Using 2D or 3D linear interpolation, find  $y_{des}$ , the lateral foot placement position corresponding to  $L_{des}$ .

#### A. Euler Integration

Euler integration, also known as the Euler method, is a numerical technique for approximating the solution of an Ordinary Differential Equation (ODE) with an initial value problem [23]. The Euler method proceeds by discretizing the time domain into small steps and approximating the solution at each step. The iterative formula for Euler integration is:

$$x_{n+1} = x_n + \Delta t \cdot f(t_n, x_n) \quad (1)$$

where  $x_{n+1}$  is the approximation of the solution at the next time step,  $x_n$  is the solution at the current time step,  $\Delta t$  is the step size representing the time interval between consecutive steps,  $f(t_n, x_n)$  is a given first-order ODE function evaluated at time  $n$ , and  $t_n$  is the current time.

The method starts with an initial value  $x_0$  and  $t_0$ , and subsequent values are computed using the iterative formula. The smaller the step size  $\Delta t$ , the more accurate the approximation, but it also increases the computational cost.

In our application, we determine our initial values  $x_0$  and  $t_0$  as being the current state of the system  $x_0 = [\theta_c(t_0) \ L(t_0)]^T$  at time  $t_0$ . We use Equation (1) to compute the state of the system at every time step  $\Delta t$  until the end of the **first** step of a period length  $t = T$ , **before** impact, where  $f(t_n, x_n)$  is defined as the system dynamics from the new ALIP model from [1]. We use an impact map (derived in Section IV-B) to compute the system dynamics **after** impact of the **first** step. These state values are equivalent to the state values at the beginning of the **next** step. Using these post-impact values as initial values, we then implement Euler integration again to compute the state of the system at every time step from the beginning of the **next** step until the end of the step **before** impact.

#### B. Impact Map for the New ALIP Model

To compute the state of the system **after** impact of the **first** step, we develop an impact map for the new ALIP model. This new impact model for the reduced order model is separate from the impact map for the full-order Cassie model.

Figure 4 illustrates the relevant variables to derive the new impact map. Let Point A be the position where the swing foot touches the ground and Point B be the point where the stance foot touches the ground. Using the angular

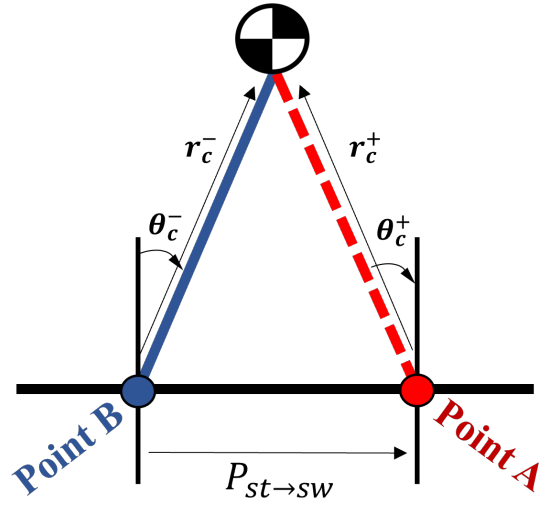


Fig. 4. Schematic of the inverted pendulum to derive the impact map for the new variation on the ALIP model.

momentum about Point B before impact  $L_B^-$ , we can compute the angular momentum about Point A after impact  $L_A^+$ . From the conservation of angular momentum, we know that the angular momentum about Point A before impact  $L_A^-$  is equivalent to the angular momentum about Point A after impact  $L_A^+$ . That is,

$$L_A^- = L_A^+ \quad (2)$$

Using the *angular momentum transfer formula* [14], we compute  $L_B^-$  from  $L_A^-$ ,

$$L_A^- = L_B^- + P_{st \rightarrow sw} \times mv_c \quad (3)$$

where  $P_{st \rightarrow sw} = [P_{st \rightarrow sw}^x \ P_{st \rightarrow sw}^z]^T$  is the vector from the stance to swing foot,  $m$  is the mass of the robot, and  $v_c$  is the CoM velocity. In the x-z plane, the position of the CoM is defined as

$$p_c = r_c^- \begin{bmatrix} \sin(\theta_c^-) \\ \cos(\theta_c^-) \end{bmatrix} \quad (4)$$

Taking the derivative Equation (4) yields the velocity of the CoM,

$$\frac{d}{dt} p_c = v_c = \begin{bmatrix} r_c^- \cos(\theta_c^-) \cdot \dot{\theta}_c^- + \dot{r}_c^- \sin(\theta_c^-) \\ -r_c^- \sin(\theta_c^-) \cdot \dot{\theta}_c^- + \dot{r}_c^- \cos(\theta_c^-) \end{bmatrix} \quad (5)$$

From Equation (2), Equation (3), and Equation (5), we compute

$$L_A^- = L_A^+ = L_B^- + m \begin{bmatrix} P_{st \rightarrow sw}^z \left( r_c^- \cos(\theta_c^-) \dot{\theta}_c^- + \dot{r}_c^- \sin(\theta_c^-) \right) - \\ P_{st \rightarrow sw}^x \left( -r_c^- \sin(\theta_c^-) \dot{\theta}_c^- + \dot{r}_c^- \cos(\theta_c^-) \right) \end{bmatrix} \quad (6)$$

Using trigonometry, we determine the value of the CoM angle after impact  $\theta_c^+$  to be

$$\theta_c^+ = \arccos \left( \frac{r_c^- \cos \theta_c^- - P_{st \rightarrow sw}^z}{r_c^+} \right). \quad (7)$$

Equation (6) and Equation (7) form our impact map. That is, the state of the system after impact is defined as

$$x^+ = \begin{bmatrix} \theta_c^+ \\ L^+ \end{bmatrix} = \begin{bmatrix} \arccos \left( \frac{r_c^- \cos \theta_c^- - P_{st \rightarrow sw}^z}{r_c^+} \right) \\ L_B^- + m \begin{bmatrix} P_{st \rightarrow sw}^z \left( r_c^- \cos(\theta_c^-) \dot{\theta}_c^- + \dot{r}_c^- \sin(\theta_c^-) \right) \\ -P_{st \rightarrow sw}^x \left( -r_c^- \sin(\theta_c^-) \dot{\theta}_c^- + \dot{r}_c^- \cos(\theta_c^-) \right) \end{bmatrix} \end{bmatrix}. \quad (8)$$

### C. Linear Interpolation

We can compute the desired lateral foot placement  $y_{des}$  by

$$y_{des} = y_{st \rightarrow sw}^1 + \frac{L_{des} - L_1}{\text{slope}}, \quad (9)$$

where

$$\text{slope} = \frac{L_2 - L_1}{y_{st \rightarrow sw}^2 - y_{st \rightarrow sw}^1}. \quad (10)$$

Similarly, one could use CoM angle,  $\theta$ , instead of angular momentum to compute lateral foot placement. That is,

$$y_{des} = y_{st \rightarrow sw}^1 + \frac{\theta_{des} - \theta_1}{\text{slope}}, \quad (11)$$

where

$$\text{slope} = \frac{\theta_2 - \theta_1}{y_{st \rightarrow sw}^2 - y_{st \rightarrow sw}^1}. \quad (12)$$

## V. NAVIGATING THE SIM-TO-REAL GAP FOR CASSIE HARDWARE

A paramount concern in deploying controllers on Cassie hardware is adhering to the stringent maximum controller execution time of 2 kHz. The passivity-based controller (PBC) detailed in [1] adeptly operates within this temporal constraint. In contrast, the model predictive controller (MPC) also discussed in [1] demands a higher execution time. To surmount this challenge, we strategically offloaded the MPC computations to a secondary computer. The execution time was further optimized through the utilization of CasADi, an open-source tool for nonlinear optimization and algorithmic differentiation [24], [25]. The MPC computation now runs at an approximate duration of 500  $\mu$ s. While it was not strictly necessary to further reduce the MPC computation time when we offloaded it to a secondary computer, it is nevertheless good practice to optimize the code whenever possible. Given that it is unnecessary to compute a desired ankle torque value for every time step, the utilization of a secondary computer for MPC, with a slower execution time dedicated to ankle torque values, poses no hindrance. This claim is substantiated by the positive outcomes observed in SimMechanics simulation experiments, where the transfer of

ankle torque computations to the secondary computer did not result in any noticeable deviations in results.

To address the observed spikes in ankle torque values during the SimMechanics simulation, as discussed in [1], we developed an improved impact map based on the linearization of the full-order impact map about the nominal trajectories. This enhancement reduced unexpected spikes in control output, further enhancing the stability and performance of Cassie in the hardware experiments discussed in this paper.

## VI. HARDWARE RESULTS AND DISCUSSION

This section delves into the practical implementation of the controllers and strategies outlined in [1] and Sections III, IV, and V on the physical 20 DOF Cassie bipedal robot hardware. While the gait demonstrations of [1] in the SimMechanics simulation environment were promising, the pivotal challenge lies in bridging the gap between simulation and reality.

### A. Maximum Walking Speed on Various Inclined Surfaces

In evaluating the performance of Cassie equipped with the novel controller structure developed in this paper, a series of tests were conducted to determine its maximum walking speed on diverse inclinations. Previous walking controllers for Cassie hardware demonstrated speeds of 2 m/s on level ground [14], but exploration of maximum speeds on inclined surfaces was lacking.

With the implementation of our new controller, Cassie showcased an enhanced walking capability, achieving a comfortable speed of up to 2.2 m/s on a flat treadmill [26]. Extending the analysis to inclined surfaces, Cassie maintained a speed of 2 m/s on a treadmill inclined at 4 degrees [27] and 8 degrees [28]. Even on steeper inclinations, such as a 15- and 20-degree slope, Cassie exhibited a walking speed, reaching up to 1.5 m/s [29], [30]. These findings underscore the adaptability and improved performance achieved through the integration of the developed controller structure, opening avenues for enhanced mobility on surfaces with varying degrees of incline.

### B. Continuous Walking on Changing Incline

In our subsequent experiment, we conducted a continuous walking test on a dynamically changing incline. The experiment commenced on a level surface with a 0-degree incline, progressively transitioning to the maximum incline of 20 degrees, and then returning to the initial 0-degree incline, as depicted in [31]. Throughout the experiment, the treadmill maintained a constant speed of 0.9 m/s. To streamline the complexity of the experiment, we employed only three nominal trajectories:

- **Nominal Trajectory for Flat Ground Walking:** Implemented during Cassie's movement on inclines ranging from 0 to 7 degrees.
- **Nominal Trajectory for 8-Degree Incline:** Applied as Cassie traversed inclines ranging from 8 to 14 degrees.



Fig. 5. Plot of left and right ankle torque values for hardware experiment on the Cassie bipedal robot walking on an inclined moving treadmill moving at a constant speed of 0.9 m/s. The treadmill is gradually inclined from 0 degrees to its maximum incline of 20 degrees and back to 0 degrees.

- **Nominal Trajectory for 15-Degree Incline:** Utilized when Cassie navigated inclines ranging from 15 to 20 degrees.

The use of nominal trajectories on inclines for which the trajectories were not specifically optimized showcased the robustness of the controller. This experiment demonstrated that perfect trajectories for all situations are not a prerequisite for stability—for example, a nominal trajectory optimized for a 15-degree incline successfully operated on a 20-degree inclined surface.

Moreover, all nominal trajectories assumed a constant nominal speed of 0.5 m/s, introducing additional disturbance to the system since the treadmill operated at 0.9 m/s during the experiment. This underscored the controller’s ability to maintain stability and adjust seamlessly to varying walking speeds, irrespective of the speed specified in the nominal trajectory. The experimental configuration provided a comprehensive assessment of the developed controller’s capacity to adapt and ensure stability during continuous walking on inclines.

Figure 5 provides insight into the left and right ankle torque values during this experiment. Importantly, the ankle torque values consistently remain well below the maximum and minimum thresholds of  $\pm 23$  N, as allowed by the Quadratic Program (QP) for the MPC.

### C. Transitioning from Stationary Flat Ground to Inclined Moving Treadmill

In our subsequent experiment, we tasked the Cassie bipedal robot with walking from stationary flat ground to an inclined moving treadmill. This challenging scenario was systematically tested for inclined transitions at 4 degrees [27], 8 degrees [28], 15 degrees [29], and 20 degrees [30]. Figure 6 shows outtakes for the experiment of the transition from stationary flat ground to a 20-degree incline. These experiments presented two significant sources of disturbance, both of which our controller adeptly surmounted.

The first challenge involved negotiating the large gaps between the stationary ground and the inclined moving treadmill. This transition demanded a seamless adjustment in gait and stability, which our controller effectively managed.

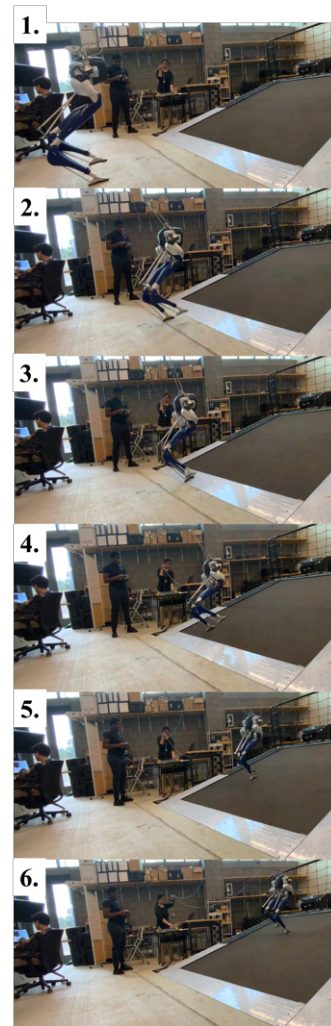


Fig. 6. A series of images depicting hardware results of the Cassie bipedal robot walking from flat, stationary ground to a 0.2 m/s moving treadmill inclined at 20 degrees from a back view perspective. Once Cassie steps on the treadmill, the treadmill’s speed is gradually increased to 1.5 m/s. The original video can be found in [30].

The second challenge arose from the fact that the treadmill was in motion at 0.2 m/s while Cassie initiated the transition from stationary flat ground to the treadmill. Despite this differential in speed, Cassie not only successfully navigated the transition but also exhibited the capability to accommodate increased treadmill speeds.

The controller demonstrated its resilience by overcoming both disturbances encountered during the transition process. Furthermore, we successfully escalated the treadmill speed to the maximum values detailed in Section VI-A post-transition, affirming the adaptability and robust performance of the developed controller in challenging real-world scenarios.

Figure 7 shows a plot of the values of the left and right ankle torque values during the experiment, from Cassie’s transition from flat stationary ground to the 20-degree inclined moving treadmill. While the ankle torque values approach the saturation threshold of  $\pm 23$  N, as allowed by the QP for the MPC, they remain below this limit,

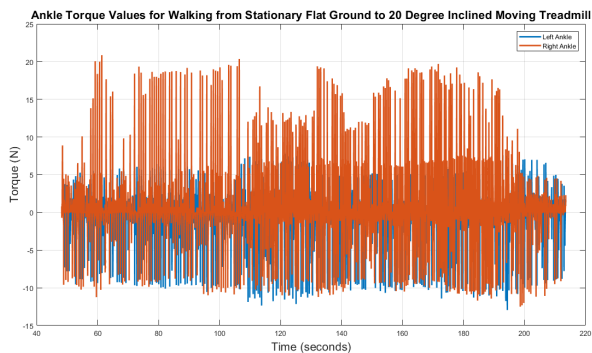


Fig. 7. Plot of left and right ankle torque values for hardware experiment on the Cassie bipedal robot walking from stationary flat ground to a 0.2 m/s moving treadmill inclined at 20 degrees. Once Cassie steps on the treadmill, the treadmill’s speed is gradually increased to 1.5 m/s.

underscoring the controller’s effectiveness in managing the complexities of dynamic transitions.

#### D. Navigating on and off of a Moving Walkway

In our culminating experiment, we tasked Cassie with traversing on and off a moving walkway—an essential capability for robots in human-centric environments such as airports and shopping malls. The integration of robots into everyday scenarios necessitates their adept navigation through commonplace features like moving walkways, which typically exhibit average speeds ranging from 0.5 to 0.83 m/s [32]. Our objective was to validate Cassie’s ability to seamlessly adapt to different speeds on the moving walkway. We successfully tested Cassie’s performance on moving walkways operating at speeds of 0.5 m/s [33], 0.8 m/s [34], and 1.2 m/s [35].

A notable challenge introduced during this experiment was the presence of a gantry, which did not extend far enough to cover the entire length of the treadmill used to simulate the moving walkway. As a result, Cassie experienced significant tugging forces from the gantry. During the initial setup, when Cassie was walked to the end of the treadmill, it stumbled and nearly fell due to the tugging forces but swiftly recovered [35]. Furthermore, despite operator error leading to Cassie approaching the moving walkway at a diagonal for the 0.8 m/s and 1.2 m/s scenarios, Cassie adeptly navigated onto and off the moving walkway. This demonstration underscores the robustness and adaptability of our controller in addressing real-world challenges during robot locomotion on dynamic surfaces.

Figure 8 provides insight into the left and right ankle torque values during Cassie’s traversal on and off a 1.2 m/s moving treadmill. Noteworthy spikes are evident between the 290-second and 295-second marks, corresponding to the moment when Cassie stumbled. Importantly, the ankle torque values consistently remain well below the maximum and minimum thresholds of  $\pm 23$  N, as allowed by the QP for the MPC. This observation attests to the controller’s ability to manage unexpected disturbances while maintaining stability, further validating its effectiveness in real-world scenarios.

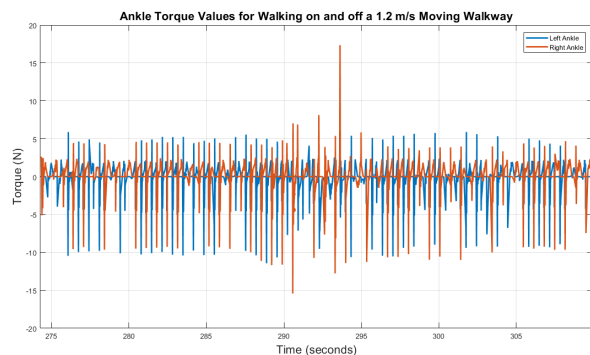


Fig. 8. Plot of left and right ankle torque values for hardware experiment on the Cassie bipedal robot walking on to and off of a 1.2 m/s moving walkway (treadmill).

## VII. CONCLUSIONS

This paper details the implementation of controllers on the physical 20 DoF Cassie bipedal robot hardware, moving beyond the commendable SimMechanics simulation outcomes of [1]. The transition to physical hardware accentuates the persistent challenge of the sim-to-real gap, encompassing issues like motor friction, sensor inaccuracies, and unforeseen variables absent in simulation environments.

The stringent maximum controller execution time of 2 kHz on Cassie hardware posed a critical concern. While the passivity-based controller (PBC) successfully adheres to this constraint, the model predictive controller (MPC) demands a longer execution time. Our strategic approach involves offloading MPC computations to a secondary computer and further optimizing execution time with CasADi. Successful simulation experiments affirm the viability of this offloading strategy.

Evaluating Cassie’s performance with the developed controller structure reveals enhanced walking capabilities. Cassie achieves speeds of up to 2.2 m/s on a flat treadmill, with maintained speeds of 2 m/s on an 8-degree inclined treadmill and 1.5 m/s on a 20-degree incline. This demonstrates the controller’s adaptability and improved performance across varying incline levels.

A continuous walking experiment on changing incline further validates the robustness of the controller. Nominal trajectories, not perfectly optimized for all situations, showcase the controller’s stability during continuous walking on inclines, emphasizing its adaptability.

Transitions from stationary flat ground to an inclined moving treadmill present unique challenges, including negotiating gaps and adapting to treadmill motion. The controller adeptly manages these disturbances, enabling successful transitions, even at increased treadmill speeds. This resilience underscores the adaptability and robust performance of the developed controller.

Cassie’s navigation on and off a moving walkway, despite challenges introduced by a tugging gantry and operator error, highlights the controller’s robustness and adaptability in real-world scenarios. This successful demonstration contributes

valuable insights for integrating bipedal robots into dynamic human-centric environments.

In conclusion, this paper demonstrates successful implementation and real-world applicability of our novel controller on Cassie hardware. By addressing challenges in execution time, hardware constraints, and adapting to dynamic surfaces, our controller contributes to the advancement of bipedal robots, fostering capabilities for enhanced mobility in diverse environments.

#### ACKNOWLEDGMENT

Funding for this work was in part provided by NSF Award No. 2118818. The work of O. Dosunmu-Ogunbi was partially supported by a Rackham Merit Fellowship.

#### REFERENCES

- [1] O. Dosunmu-Ogunbi, A. Shrivastava, G. Gibson, and J. W. Grizzle, "Stair climbing using the angular momentum linear inverted pendulum model and model predictive control," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2023, pp. 8558–8565.
- [2] M. J. Powell and A. D. Ames, "Mechanics-based control of underactuated 3D robotic walking: Dynamic gait generation under torque constraints," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. IEEE, 2016, pp. 555–560. [Online]. Available: <http://ames.caltech.edu/iros2016pa.pdf>
- [3] Y. Gong and J. Grizzle, "One-step ahead prediction of angular momentum about the contact point for control of bipedal locomotion: Validation in a lip-inspired controller," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*, 2021, pp. 2832–2838.
- [4] C. Fu and K. Chen, "Gait synthesis and sensory control of stair climbing for a humanoid robot," *IEEE Transactions on Industrial Electronics*, vol. 55, no. 5, pp. 2111–2120, 2008.
- [5] S. Caron, A. Kheddar, and O. Tempier, "Stair climbing stabilization of the hrp-4 humanoid robot using whole-body admittance control," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 277–283.
- [6] A. Hereid, O. Harib, R. Hartley, Y. Gong, and J. W. Grizzle, "Rapid trajectory optimization using c-frost with illustration on a cassie-series dynamic walking biped," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2019, pp. 4722–4729.
- [7] J. Verhagen, X. Xiong, A. Ames, and A. Seth, "From human walking to bipedal robot locomotion: Reflex inspired compensation on planned and unplanned downsteps," 2022. [Online]. Available: <https://arxiv.org/abs/2209.02995>
- [8] J. Siekmann, K. Green, J. Warila, A. Fern, and J. W. Hurst, "Blind bipedal stair traversal via sim-to-real reinforcement learning," *CoRR*, vol. abs/2105.08328, 2021. [Online]. Available: <https://arxiv.org/abs/2105.08328>
- [9] Q. Nguyen, A. Hereid, J. W. Grizzle, A. D. Ames, and K. Sreenath, "3d dynamic walking on stepping stones with control barrier functions," in *2016 IEEE 55th Conference on Decision and Control (CDC)*, 2016, pp. 827–834.
- [10] M. Dai, X. Xiong, and A. Ames, "Bipedal walking on constrained footholds: Momentum regulation via vertical com control," in *2022 International Conference on Robotics and Automation (ICRA)*, 2022, pp. 10 435–10 441.
- [11] B. Acosta and M. Posa, "Bipedal walking on constrained footholds with mpc footstep control," in *2023 IEEE-RAS 22nd International Conference on Humanoid Robots (Humanoids)*. IEEE, 2023, pp. 1–8.
- [12] X. Xiong and A. Ames, "Slip walking over rough terrain via h-lip stepping and backstepping-barrier function inspired quadratic program," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 2122–2129, 2021.
- [13] B. Griffin and J. Grizzle, "Nonholonomic virtual constraints and gait optimization for robust walking control," *The International Journal of Robotics Research*, vol. 36, no. 8, p. 895–922, Jul 2017.
- [14] Y. Gong and J. Grizzle, "Zero dynamics, pendulum models, and angular momentum in feedback control of bipedal locomotion," 2021.
- [15] G. Gibson, O. Dosunmu-Ogunbi, Y. Gong, and J. Grizzle, "Terrain-adaptive, alip-based bipedal locomotion controller via model predictive control and virtual constraints," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 6724–6731.
- [16] Y. Gong, R. Hartley, X. Da, A. Hereid, O. Harib, J.-K. Huang, and J. Grizzle, "Feedback control of a cassie bipedal robot: Walking, standing, and riding a segway," in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 4559–4566.
- [17] A. Hereid, E. A. Cousineau, C. M. Hubicki, and A. D. Ames, "3d dynamic walking with underactuated humanoid robots: A direct collocation framework for optimizing hybrid zero dynamics," in *2016 IEEE International Conference on Robotics and Automation (ICRA)*, May 2016, p. 1447–1454.
- [18] K. Sreenath, H.-W. Park, I. Poulakakis, and J. W. Grizzle, "A compliant hybrid zero dynamics controller for stable, efficient and fast bipedal walking on mabel," *The International Journal of Robotics Research*, vol. 30, no. 9, p. 1170–1193, Aug 2011.
- [19] K. A. Hamed, J. Kim, and A. Pandala, "Quadrupedal locomotion via event-based predictive control and qp-based virtual constraints," *IEEE Robotics and Automation Letters*, vol. 5, no. 3, p. 4463–4470, Jul 2020.
- [20] A. Hereid and A. D. Ames, "Frost: Fast robot optimization and simulation toolkit," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. Vancouver, BC, Canada: IEEE/RSJ, Sep. 2017.
- [21] G. Farin, "Algorithms for rational bézier curves," *Computer-aided design*, vol. 15, no. 2, pp. 73–77, 1983.
- [22] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris, *Feedback control of dynamic bipedal robot locomotion*. CRC press, 2018.
- [23] B. Biswas, S. Chatterjee, S. Mukherjee, and S. Pal, "A discussion on euler method: A review," *Electronic Journal of Mathematical Analysis and Applications*, vol. 1, no. 2, pp. 2090–2792, 2013.
- [24] 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, p. 1–36, Mar 2019.
- [25] J. A. Andersson and J. B. Rawlings, "Sensitivity analysis for nonlinear programming in casadi\*," *IFAC-PapersOnLine*, vol. 51, no. 20, pp. 331–336, 2018, 6th IFAC Conference on Nonlinear Model Predictive Control NMPC 2018. [Online]. Available: <https://www.sciencedirect.com/science/article/pii/S2405896318327137>
- [26] DynamicLegLocomotion, "Cassie walking up to 2.2 m/s on flat treadmill," <https://www.youtube.com/watch?v=MH8TWk1SjGM>, December 2023, accessed: 2023-12-15.
- [27] —, "Cassie walks from stationary flat ground to 4-degree inclined moving treadmill," <https://www.youtube.com/watch?v=ZnPoVpUDWQs>, December 2023, accessed: 2023-12-15.
- [28] —, "Cassie walks from stationary flat ground to 8-degree inclined moving treadmill," <https://www.youtube.com/watch?v=DnYR85WktXk>, December 2023, accessed: 2023-12-15.
- [29] —, "Cassie walks from stationary flat ground to 15-degree inclined moving treadmill," <https://www.youtube.com/watch?v=0DaqhGlrX38>, December 2023, accessed: 2023-12-15.
- [30] —, "Cassie walks from stationary flat ground to 20-degree inclined moving treadmill," <https://www.youtube.com/watch?v=52nT6H3xkBI>, December 2023, accessed: 2023-12-15.
- [31] —, "Continuous walking on changing incline," <https://www.youtube.com/watch?v=ZGtOIRkN1RU>, December 2023, accessed: 2023-12-15.
- [32] I. Kusumaningtyas and G. Lodewijks, "Accelerating moving walkway: A review of the characteristics and potential application," *Transportation research part A: policy and practice*, vol. 42, no. 4, pp. 591–609, 2008.
- [33] DynamicLegLocomotion, "Cassie on a 0.5 m/s moving walkway," <https://www.youtube.com/watch?v=rZZ9UYAjRvc>, December 2023, accessed: 2023-12-15.
- [34] —, "Cassie on a 0.8 m/s moving walkway," <https://www.youtube.com/watch?v=1MAp6qj8pdM>, December 2023, accessed: 2023-12-15.
- [35] —, "Cassie on a 1.2 m/s moving walkway," <https://www.youtube.com/watch?v=utANK8jTwuI>, December 2023, accessed: 2023-12-15.
- [36] —, "Cassie walks up stairs in simmechanics," <https://www.youtube.com/watch?v=TX74b2i9p-k>, December 2023, accessed: 2023-12-15.
- [37] J. M. Fitch, J. Templer, and P. Corcoran, "The dimensions of stairs," *Scientific American*, vol. 231, no. 4, pp. 82–91, 1974.