

CONTROL OF A BIPEDAL WALKING USING PARTIAL FEEDBACK LINEARIZATION AND GAUSSIAN PROCESS REGRESSION-BASED OF THE STEP-TO-STEP MAP

Daniel Torres¹, Ernesto Hernandez Hinojosa¹, Pranav A. Bhounsule¹

¹University of Illinois at Chicago, Chicago, IL

ABSTRACT

Bipedal robots with small and point feet find it difficult to balance instantaneously due to underactuation, but can be relatively easy to control over the time scale of a step. Thus, models of step-to-step dynamics provide an effective means for the development of step-to-step controllers. Such models are analytically intractable because they are obtained from integrating nonlinear equations of motion. Our hypothesis is that obtaining approximations of such models can be effective for real-time computations of control inputs. We use partial feedback linearization to reduce the system dimensions for the step-to-step map. We formulate and solve an optimal control problem with the approximations to compute control for tracking a reference velocity and walking on stepping stones. These solutions demonstrate the need for bipedal robots to achieve specific control objectives, such as walking speed and foot placement, in order to navigate through restricted environments such as the stepping stones. We provide simulations on a 5-link biped to show the efficacy of the approach. A video link is provided: <https://www.youtube.com/watch?v=p2TLU3Jzul8>.

Keywords: Control, Partial Feedback Linearization, Gaussian Process, Nonlinear Model, Poincaré Map

1. INTRODUCTION

Bipedal robots are well-suited for applications in homes and warehouses. These applications would require a robust walking gait that can be used in conjunction with task planning, path planning, motion planning, computer vision, and manipulation for use in practical applications. However, the challenge of reliable walking in simple environments has still not been accomplished.

Bipedal systems generally have point or small feet which leads to underactuation (i.e., the number of degrees of freedom exceeds the number of actuators on the degrees of freedom). Because of underactuation, such systems are nearly impossible to control instantaneously (e.g., balancing over the upright position when perturbed). However, it is possible to control these systems over a finite time horizon, typically of the order of one step [2]. For example, one can correct a push given to a standing robot

with point feet by taking a step in the direction of the push, thus achieving balance control over one step [3]. This type of control, known as step-to-step control is the most accepted method of controlling underactuated bipedal systems.

The step-to-step level control is typically solved offline with hi-fidelity physics models and adapted for real-time control through a look-up table [4]. However, this approach fails to generalize to novel situations. Another approach is to use an online method where simple models are used for computing the controller and then mapping to the hi-fidelity model during run-time [5]. This approach is often conservative because the simple models do not capture the complex dynamics. We present an approach that can be generalized to novel scenarios, but captures the hi-fidelity of the physics model while being computationally modest. First, we use Partial Feedback Linearization (PFL) to transform the complex model to low dimensions. Second, we approximate the step-to-step dynamics of the hi-fidelity model using a Gaussian Process Regression (GPR). Finally, we solve an optimization problem with GPR-based step-to-step model during run-time to compute the correct control signal.

2. BACKGROUND AND RELATED WORK

Origins of Step-to-Step Control: The step-to-step control was first presented in the context of passive dynamic walking by McGeer [6]. McGeer showed that a design that resembles a human frame when designed properly can walk down a shallow slope with no control. Then McGeer interpreted the resulting periodic motion as a limit cycle (a repeating trajectory) and used the eigenvalues of the linearization of the step-to-step map, also known as the Poincaré map (i.e., the function that maps the initial conditions from one step at a particular instant in the locomotion cycle to the same instant at the next step), to analyze the stability of the system [7]. Finally, McGeer showed that by using a linear controller based on the step-to-step map, it was possible to increase the robustness of the passive limit cycle [8].

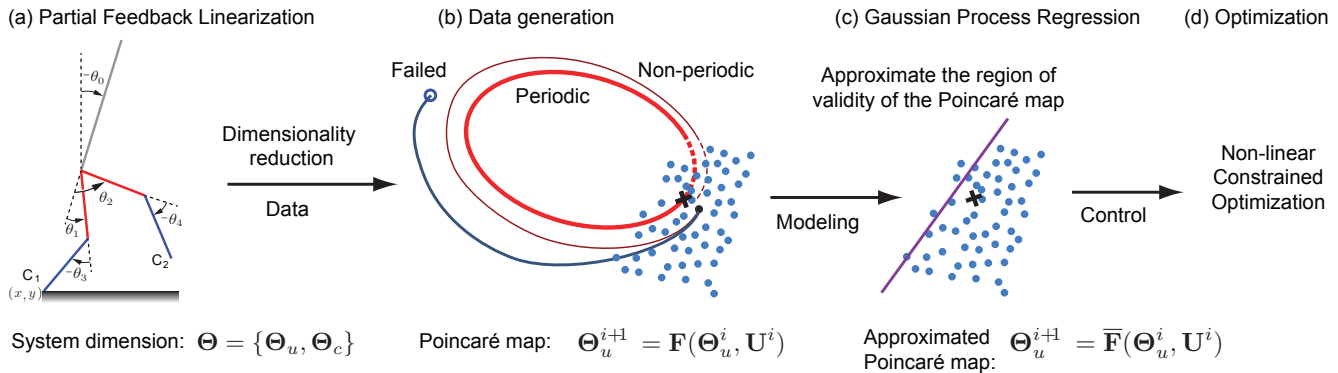


FIGURE 1: OVERVIEW OF THE APPROACH: (A) PFL REDUCES THE STANCE PHASE DYNAMICS FROM $\Theta = [\Theta_u, \Theta_c]$ (10 DIMENSIONS) TO $\Theta = \Theta_u$ (2 DIMENSIONS). (B) A POINCARÉ SECTION IS CHOSEN AT MID-STANCE. WE GENERATE RANDOM INPUT STATE AT THE POINCARÉ SECTION AND CONTROLS AT THE STEP AND SIMULATE TILL THE NEXT POINCARÉ SECTION TO GENERATE DATA FOR THE POINCARÉ MAP GIVEN BY F , $\Theta_u^{i+1} = F(\Theta_u^i, U^i)$. (C) THE POINCARÉ MAP IS CURVE FITTED $\Theta_u^{i+1} = \bar{F}(\Theta_u^i, U^i)$ WHERE \bar{F} IS A GAUSSIAN PROCESS REGRESSION MODEL AND GAUSSIAN PROCESS CLASSIFICATION IS USED TO IDENTIFY THE BOUNDARY OF THE MODEL. (D) NONLINEAR PROGRAMMING IS USED TO SOLVE A SUITABLY FORMULATED QUADRATIC PROGRAM. FOR A VIDEO SEE [1].

Approximations to the Step-to-Step Map: McGeer’s approach to linearize the Poincaré map achieves stability in the small neighborhood of the limit cycle (periodic gait). To achieve a large range of stability, there have been attempts to compute the nonlinear approximation to the step-to-step map.

This is a data-driven approach that relies on a simulator to approximate the step-to-step map. The idea is to sample state and control (or state-action) pairs at the Poincaré section and integrate the equations to the next Poincaré section. This provides data for the step-to-step map which can then be curve fitted to a nonlinear function. The step-to-step map of a springy quadruped has been fitted with a neural network and then a model-predictive controller has been used for control [9]. Our past work considered using polynomial, neural network, and GPR to fit the step-to-step map of a spring-loaded inverted pendulum and the use of trajectory optimization for control. The neural network and GPR outperformed the polynomial regression [10]. One can also use the step-to-step map approximation of simple models to control more complex robots [11]. The step-to-step map may also use data from hardware experiments to obtain a better approximation [12].

Approximations to the Region of Validity of the Step-to-Step Map: One issue with these approaches is that not all state-action input pairs to the step-to-step map lead to a successful step; there might be inputs that lead to failure. Hence, it is important to model the boundary of the feasibility of the step-to-step map. Our past work used support vector machines to model the region of validity of the step-to-step map [13]. Another approach found that neural networks can be used for modeling the step-to-step map [14].

Model Reduction Over Step-to-Step Map: Although bipeds are high-dimensional systems with underactuation, one can use feedback control to reduce the dimensionality of the system.

The hybrid zero dynamics (HZD) approach is a dimensionality reduction technique [15, 16]. Here one defines continuous-time outputs (also known as virtual constraints) that map the actuated degrees of freedom to the unactuated degrees of freedom. One then designs a controller to drive these outputs to zero. This method is attractive because it uses control to reduce the dimensionality of the system to the unactuated degrees of freedom and is scalable. As the system complexity increases, it is not very easy to find these virtual constraints that lead to acceptable performance [17]. Although one can create exponentially stable continuous-time controllers, the orbital stability is only asymptotically stable [18].

In this work, we capitalize on the use of partial feedback linearization (PFL) to reduce the dimensionality of the system to the uncontrolled degree of freedom at the Poincaré map [13]. The system considered here has N actuators and $N + 1$ degrees of freedom. We use PFL to do continuous control of N degrees of freedom using the N actuators. However, these N actuator commands have free parameters. These free parameters can then be used to control the step-to-step map achieving control of the 1 unactuated degree of freedom at the time scale of a step.

The organization of the paper is as follows. In Sec. 3 we provide details of the physics-based model for the 5-link biped. In Sec. 4 we provide the methods for reducing the dimensionality of the system, approximating the Poincaré map, and formulating the non-linear constrained program. In Sec. 5 we present the results of the models and optimizations. Sec. 6 is the Discussion and Sec. 7 is the Conclusions and Future Work.

3. MODEL

Figure 2 shows the 2D, 5-link model used in this study. This model and ensuing write-up is from our past work [13]. We define the stance leg as the one in contact with the ground and the swing leg as the other. The foot in contact with the ground has coordinates (x, y) where the x-axis is horizontal and y-axis is vertical. The torso angle θ_0 is the angle between the torso and

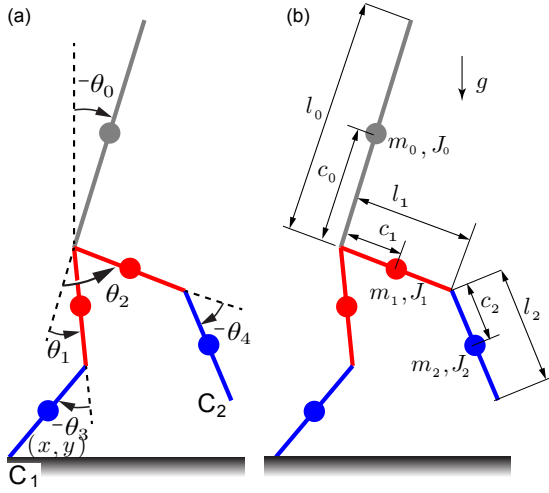


FIGURE 2: HUMANOID MODEL: (A) CONFIGURATION VARIABLES DESCRIBING THE DEGREES OF FREEDOM, (B) MASS, CENTER OF MASS, INERTIA ABOUT CENTER OF MASS, AND LENGTH PARAMETERS.

the vertical direction, θ_1 and θ_2 are the relative angles made by the thigh links of the stance and swing leg (respectively with the torso), and θ_3 and θ_4 are the angles made by the calf links of the stance and swing leg with their respective thigh links. The torso mass is $m_0 = 50$ kg, center of mass is at $c_0 = 0.5$ m, and inertia about the center of mass is $J_0 = 10$ kg-m². The thigh links have a mass of $m_1 = 7$ kg, center of mass is at $c_1 = 0.25$ m, and inertia about the center of mass is $J_1 = 5$ kg-m². The calf links have a mass of $m_2 = 5$ kg, center of mass at $c_2 = 0.25$ m, and inertia about the center of mass is $J_2 = 2$ kg-m². Gravity points downwards and is $g = 9.81$ m/s². The torso length $\ell_0 = 1$ m the thigh link and calf link lengths are equal, $\ell_1 = \ell_2 = 0.5$.

There are two sets of equations: one for the single stance phase where one foot is on the ground and the second for the foot-strike where the legs exchange roles.

3.1 Single Stance Equations

The state variables for derivation are defined as $\mathbf{q} = [x \ y \ \theta_0 \ \theta_1 \ \theta_2 \ \theta_3 \ \theta_4]^T$. We include the floating coordinates x and y to derive the equation, but the simplified equation has only five variables: $\theta_0, \theta_1, \dots, \theta_4$. The Lagrangian $\mathcal{L} = \mathcal{T} - \mathcal{V} = 0.5 \sum (m_i v_i^T v_i + J_i \omega_i^T \omega_i) - \sum (m_i g y_i)$ where v_i, ω_i, y_i are the linear velocity, angular velocity, and y-position center of mass of link i , respectively. We take the summation over all the five links. Using the Euler-Lagrange equations gives seven equations:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{N}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{B}\mathbf{u} + \mathbf{J}_{C_1}\mathbf{P}_{C_1} \quad (1)$$

where $\mathbf{M}, \mathbf{N}, \mathbf{B}$ are the mass matrix, accelerations due to Coriolis, centrifugal acceleration and gravity, and torque selection matrices, respectively. The control torques are $\mathbf{u} = [\tau_1 \ \tau_2 \ \tau_3 \ \tau_4]^T$ where τ_i is the torque for joint with stance calf link θ_i . The Jacobian is \mathbf{J}_{C_1} of the contact point C_1 and \mathbf{P}_{C_1} is the ground reaction force on the stance leg.

Without loss of generality, we can assume $x = y = 0$. Also, since C_1 is at rest, $\dot{x} = \dot{y} = \ddot{x} = \ddot{y} = 0$. Using these conditions, we use the first two equations in Eqn. 1 to find the ground reaction forces \mathbf{P}_{C_1} as a function of joint angles, velocities, and acceleration. We may write the remaining five equations as:

$$\mathbf{M}_\theta(\theta)\ddot{\theta} + \mathbf{N}_\theta(\theta, \dot{\theta}) = \mathbf{B}_\theta\mathbf{u} \quad (2)$$

where $\mathbf{M}_\theta, \mathbf{N}_\theta, \mathbf{B}_\theta$ are versions of the matrices defined earlier. We use this equation for simulating single stance phase and for controller development later.

3.2 Foot-Strike Equations

When the swing foot C_2 touches the ground, the single stance phase ends and the robot transitions to an instantaneous foot-strike phase. We assume that the trailing leg applies an impulsive force along the stance leg, $\mathbf{I}_{C_1} = I [-\sin(\theta_0 + \theta_1 + \theta_3), \cos(\theta_0 + \theta_1 + \theta_3)]^T$ where I is the scalar impulse. This force comes from the ankle motor at C_1 which is passive during the stance phase, except during the foot-strike phase. Our choice of impulsive push-off is to be able to achieve energy-efficient walking compared to hip actuation (see [19]). In this phase, angular momentum is conserved about new contact point C_2 . We obtain the equations for this phase by integrating Eqn. 1 and taking the limit as time goes to 0:

$$\begin{bmatrix} \mathbf{M}(\mathbf{q}^-) & -\mathbf{J}_{C_2}^T \\ \mathbf{J}_{C_2} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{q}^+ \\ \mathbf{I}_{C_2} \end{bmatrix} = \begin{bmatrix} \mathbf{M}(\mathbf{q}^-)\dot{\mathbf{q}}^- + \mathbf{J}_{C_1}^T \mathbf{I}_{C_1} \\ \mathbf{0} \end{bmatrix} \quad (3)$$

where the superscript $-$ and $+$ denote the instance before and after collision, respectively.

3.3 Simulating a Single Step

Figure 3 shows the general equation that describes a single step, the repeating unit, that starts and ends at mid-stance. We now explain the composition of a single step. We start the step at mid-stance when stance leg thigh link is vertical, $\theta_0 + \theta_1 = 0$. Next, we use the single stance Eqn. 2 to integrate the system till foot-strike. The foot strike occurs when the swing foot C_2 touches the ground, $y_{C_2} = \ell_1 \cos(\theta_0 + \theta_1) - \ell_1 \cos(\theta_0 + \theta_2) + \ell_2 \cos(\theta_0 + \theta_1 + \theta_3) - \ell_2 \cos(\theta_0 + \theta_2 + \theta_4) = 0$. Next, we apply the foot-strike condition given by Eqn. 3. Then we swap the legs, $\theta_0^+ = \theta_0^-$, $\theta_1^+ = \theta_2^-$, $\theta_2^+ = \theta_1^-$, $\theta_3^+ = \theta_4^-$, $\theta_4^+ = \theta_3^-$. Similarly, for the angular velocities we have $\dot{\theta}_0^+ = \dot{\theta}_0^-$, $\dot{\theta}_1^+ = \dot{\theta}_2^-$, $\dot{\theta}_2^+ = \dot{\theta}_1^-$, $\dot{\theta}_3^+ = \dot{\theta}_4^-$, $\dot{\theta}_4^+ = \dot{\theta}_3^-$. Finally, we integrate the equations in single stance given by Eqn. 2 till the next mid-stance given by $\theta_0 + \theta_1 = 0$.

4. METHODS

4.1 Partial Feedback Linearization

PFL is utilized to regulate the actuated degrees of freedom during the stance phase. The mass matrix inversion of Eqn. 2 results in:

$$\ddot{\theta} = \mathbf{M}_\theta^{-1}(\theta)(\mathbf{B}_\theta\mathbf{u} - \mathbf{N}_\theta(\theta, \dot{\theta})) \quad (4)$$

The system is underactuated as it only has four actuators to control the five degrees of freedom. PFL is utilized to decouple the following degrees of freedom: the torso θ_0 , the swing leg

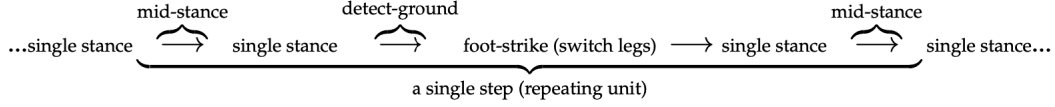


FIGURE 3: A REPRESENTATION OF THE WALKING CYCLE.

joints θ_2 and θ_4 , and the stance leg knee θ_3 . The decoupled degrees of freedom can be assigned as $\theta_c = [\theta_0 \ \theta_2 \ \theta_3 \ \theta_4]$. A matrix \mathbf{S}_c can be found with the relation $\theta_c = \mathbf{S}_c\theta$, where $\mathbf{S}_c = \text{diag}\{1, 0, 1, 1, 1\}$ and $\theta = [\theta_0 \ \theta_1 \ \theta_2 \ \theta_3 \ \theta_4]$. Taking this relation with Eqn. 4 results in:

$$\ddot{\theta}_c = \mathbf{S}_c\ddot{\theta} = \mathbf{S}_c\mathbf{M}_\theta^{-1}(\theta)(\mathbf{B}_\theta\mathbf{u} - \mathbf{N}_\theta(\theta, \dot{\theta})) = \mathbf{v} \quad (5)$$

where \mathbf{v} is the control input:

$$\mathbf{v} = \ddot{\theta}_c^{ref} + \mathbf{K}_d(\dot{\theta}_c^{ref} - \dot{\theta}_c) + \mathbf{K}_p(\theta_c^{ref} - \theta_c) \quad (6)$$

Reference position θ_c^{ref} , velocity $\dot{\theta}_c^{ref}$, and acceleration $\ddot{\theta}_c^{ref}$ are assigned. A fifth order polynomial is selected with initial and final position, velocity, and acceleration specified with most set to 0. \mathbf{K}_p and \mathbf{K}_d are diagonal matrices, with $\mathbf{K}_p = K_p \text{diag}\{1, 1, 1, 1\}$ and $\mathbf{K}_d = 2\sqrt{K_p} \text{diag}\{1, 1, 1, 1\}$, chosen to achieve critical damping. The motor torques are computed as follows:

$$\mathbf{u} = (\mathbf{S}_c\mathbf{M}_\theta^{-1}(\theta)\mathbf{B}_\theta)^{-1}(\mathbf{v} + \mathbf{S}_c\mathbf{M}_\theta^{-1}(\theta)(\mathbf{N}_\theta(\theta, \dot{\theta}))) \quad (7)$$

By defining the uncontrolled degree of freedom as $\theta_u = \mathbf{S}_u\theta$, where $\mathbf{S}_u = \text{diag}\{0, 1, 0, 0, 0\}$ and $\theta_u = \theta_1$, we can formulate an equation for this degree of freedom, incorporating the appropriate control input from the equation above:

$$\ddot{\theta}_u = \mathbf{S}_u\ddot{\theta} = \mathbf{S}_u\mathbf{M}_\theta^{-1}(\theta)(\mathbf{B}_\theta\mathbf{u} - \mathbf{N}_\theta(\theta, \dot{\theta})) \quad (8)$$

This equation is integrated in the single stance phase.

4.2 Step-to-Step Dynamics: Poincaré Map

The Poincaré section is a surface of $2N - 1$ dimensions (with N being the total degrees of freedom of the system) that represents a particular instance in the locomotion cycle, such as mid-stance or foot-strike. A Poincaré section is depicted as blue dots in Fig. 1 (b). The Poincaré map is a function, denoted as \mathbf{F} , mapping the initial state at the Poincaré section Θ^i and the controls during the step \mathbf{U}^i to the state at the next Poincaré section Θ^{i+1} . This map \mathbf{F} describes the step-to-step dynamics and is determined by integrating equations from successive Poincaré sections shown in Fig. 3. Refer to [20] for more details on the Poincaré section and map. The Poincaré section is written as:

$$\Theta^{i+1} = \mathbf{F}(\Theta^i, \mathbf{U}^i) \quad (9)$$

where i represents the step number, the state is represented by $\Theta = [\theta \ \dot{\theta}]$, and θ is defined as $[\theta_0 \ \theta_1 \ \theta_2 \ \theta_3 \ \theta_4 \ \theta_5]$. The discrete controls are denoted by \mathbf{U} , foot placement angle and impulsive push-off, and are set once per step. The Poincaré map, \mathbf{F} , is chosen to map the successive mid-stance states. Generally, the Poincaré map is determined numerically as it is not

always feasible to obtain an analytical formula. To approximate the Poincaré map, the equations of motion are integrated and/or the algebraic conditions for instantaneous phases (such as foot-strike) are applied. Mid-stance is defined as $\theta_0 + \theta_1 = 0$. The Poincaré map is nine dimensional for this system with 10 degrees of freedom.

If the PFL is assumed to function as intended, the step-to-step dynamics rely solely on the uncontrolled degrees of freedom, which can be represented as $\Theta = [\theta_u \ \dot{\theta}_u]$. For the 5-link biped, θ_u is equivalent to θ_1 . However, there is only one degree of freedom, $\dot{\theta}_1$, since the Poincaré map occurs at mid-stance. The controls are chosen as the step angle and push-off at foot-strike, denoted as $\theta_2 = \alpha$ and I , respectively. The system can be expressed as:

$${}^m\dot{\theta}_1^{i+1} = F({}^m\dot{\theta}_1^i, \alpha^i, I^i) \quad (10)$$

where ${}^m\dot{\theta}_1^i$ is the mid-stance speed of θ_1 .

4.3 Poincaré Map Approximation

An analytical solution for the step-to-step dynamics is not feasible. A simple approximation \bar{F} must be obtained where ${}^m\dot{\theta}_1^{i+1}$ is the estimated mid-stance velocity at step $i + 1$:

$${}^m\dot{\theta}_1^{i+1} = \bar{F}({}^m\dot{\theta}_1^i, \alpha^i, I^i) \quad (11)$$

4.3.1 Data Generation. A simulator is designed as shown in Fig. 3 where the mid-stance speed ${}^m\dot{\theta}_1^i$, the foot placement angle α^i , and the push-off impulse I^i are the inputs and the mid-stance speed at the next step ${}^m\dot{\theta}_1^{i+1}$ is the output. While some input combinations can result in a feasible step, others may cause the model to lose balance and fall before taking the next step, rendering them infeasible. The data set consists of 1000 combinations of $\dot{\theta}_1^i, \alpha^i, I^i$, and has been divided into training and testing sets with an 80-20 split. The entire training data set has been used to fit a Gaussian Process Classification (GPC) model to predict the feasibility of a step based on its state-action pair, and the data that results in a feasible step is utilized to fit a GPR model for the 2-BVP problem.

4.3.2 Predicting Feasibility of State-Action Pairs. A GPC model is used to predict whether a state-action pair will result in a feasible step. A GPR model is created with the MATLAB fitrgp function, an automatic relevance determination (ARD) squared exponential kernel, and expected improvement acquisition function. The hyper-parameters, inputs, and outputs are:

$$L = [\sigma({}^m\dot{\theta}_1^i) \ \sigma(\alpha^i) \ \sigma(I^i)] \quad (12)$$

$$\sigma_n = \sigma(y) \quad (13)$$

$$X = [{}^m\dot{\theta}_1^i \ \alpha^i \ I^i] \quad (14)$$

$$y_{GPC} = \text{feasibility} \in [0, 1] \quad (15)$$

where all inputs X and outputs y_{GPC} pertain to the training data set. The hyper-parameters L and σ_n represent the length scales of the ARD squared exponential (one for each input dimension) and the signal standard deviation. The σ function is the standard deviation of each of the inputs. The intermediary GPR function is converted to a GPC by comparing the output with a threshold value, ϵ , to split the data into two classes, feasible and infeasible. The GPC model is given by:

$$y_{GPC} = \begin{cases} 0 \text{ (infeasible)} & \text{if } GPC(X) < \epsilon \\ 1 \text{ (feasible)} & \text{if } GPC(X) \geq \epsilon \end{cases} \quad (16)$$

where $\epsilon = 0.3$.

4.3.3 Gaussian Process Regression Model of the Poincaré Map. After developing the classification model, the feasible samples were used to fit a GPR model for the Poincaré map. The GPR hyper-parameters and inputs were formulated with Eqns. 12 - 14. The GPR model is:

$$y_{GPR} = GPR(X) \quad (17)$$

where $y_{GPR} = \theta_A^{i+1}$.

4.4 Quadratic Program

An optimization problem was formulated to find the inputs that approximate θ_A^{i+1} . The problem can be written as:

$$\underset{\mathbf{x}}{\text{minimize}} \quad f(\mathbf{x}) \quad (18)$$

$$\text{subject to: } g(\mathbf{x}) = 0 \quad (19)$$

$$h(\mathbf{x}) < 0 \quad (20)$$

$$\mathbf{LB} \leq \mathbf{x} \leq \mathbf{UB} \quad (21)$$

where $\mathbf{x} = [\Theta^{i+1} \quad \mathbf{U}^i]^T$. The cost function is $f(x) = ({}^m\theta_1^{i+1} - m\bar{\theta}_1^{i+1})^2 + (\alpha^i - \bar{\alpha}^i)^2 + (I^i - \bar{I}^i)^2$ where $m\bar{\theta}_1^{i+1}$, $\bar{\alpha}^i$ and \bar{I}^i have the nominal values -0.97 , 0.375 , 0.18 , respectively. The nominal values pertain to walking at human speed and step length [21]. The equality constraint, Eqn. 19, is given by $g(x) = GPR(x) - \theta_A^{i+1}$. The inequality constraint, Eqn. 20, is given by $h(x) = GPC(x) + \epsilon$. \mathbf{LB} and \mathbf{UB} are the lower and upper bound limits set to $[-2.5, 0.087, 0.1]$ and $[-0.5, 0.873, 0.4]$, respectively. The bounds were chosen to contain the training data range and extend beyond. This allows for some flexibility in selecting the control parameters and applying them for different activities such as walking in a constrained environment.

5. RESULTS

5.1 Gaussian Process Classification

Figure 4 shows the feature space of the test set where the red and blue dots are the infeasible and feasible samples, respectively. The confusion matrix of the GPC model is shown in Fig. 5. Overall, the model has an accuracy of 99.5%. The model predicts infeasible data (0) and feasible data (1) with 100% and 98.2% accuracy, respectively.

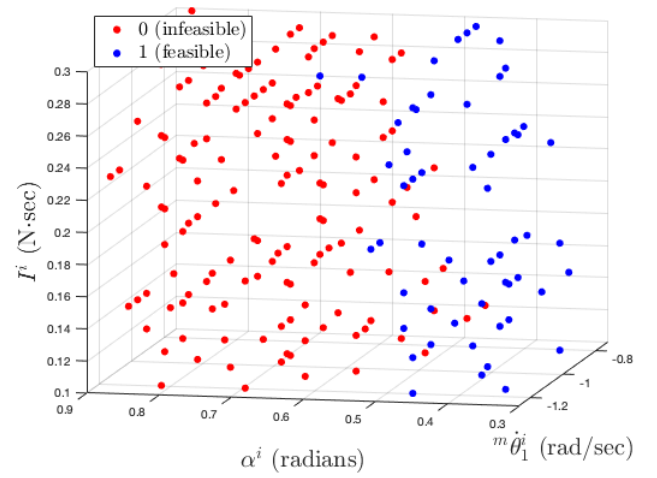


FIGURE 4: FEATURE SPACE OF THE TEST SET.

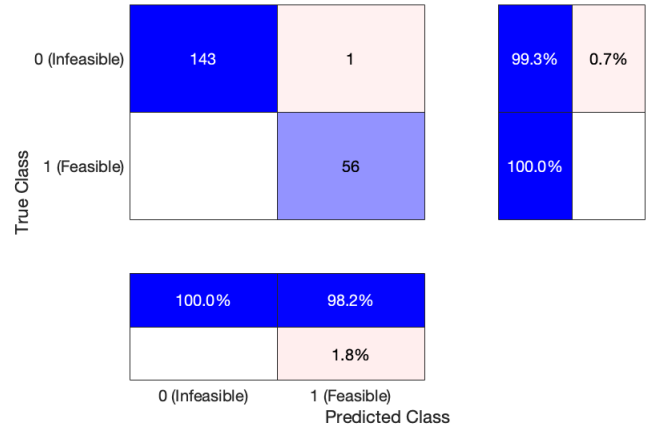


FIGURE 5: CONFUSION MATRIX FOR THE TEST SET.

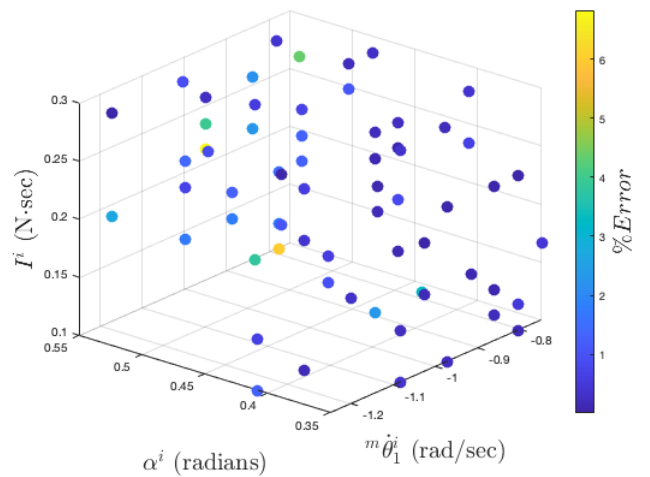


FIGURE 6: GPR PERFORMANCE FOR THE TEST SET.

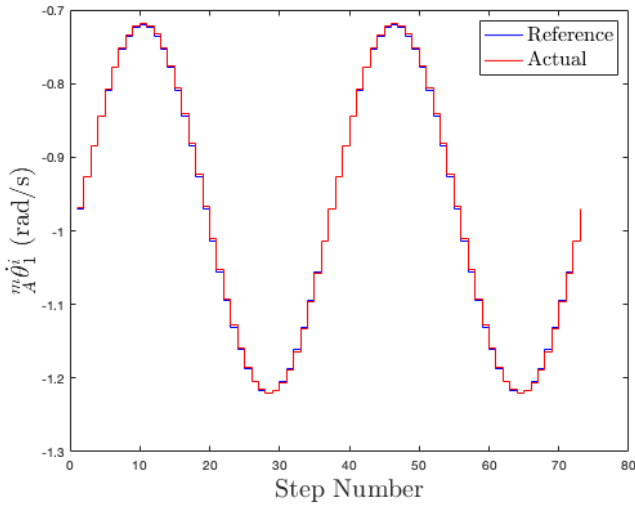


FIGURE 7: SINUSOIDAL VELOCITY TRACKING PERFORMANCE.

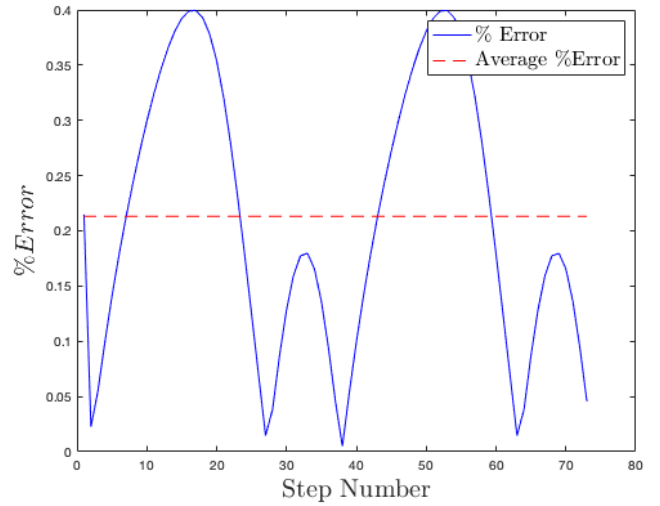


FIGURE 8: VELOCITY TRACKING PERCENTAGE ERROR.

5.2 Gaussian Process Regression

Figure 7 shows that the GPR model was incorrect by at most 7% or 93% accurate. Taking into consideration the nominal value of -0.97 rad/s, the GPR model predicted 67.7% of the data with a 99% accuracy and predicted 96.9% of the data with a 95% accuracy. The average accuracy of the model was 98.96%.

5.3 Optimization: Velocity Tracking

The model and optimization framework were tested by tracking a sinusoidal velocity profile ($\hat{\theta}_1^{ref}$) for 35 steps. The reference profile had an offset equivalent to the nominal velocity and an amplitude of 0.25 rad/sec. The cost function was modified to account for the reference profile by changing $m\hat{\theta}_1^{i+1}$ to $\hat{\theta}_1^{ref}$. The performance of the optimization solution with respect to the reference velocity is shown in Fig. 7. The percent error in tracking the reference velocity is shown in Fig. 8. The average percent error is 0.21% and shown in red. The number of function evaluations ranged from 12 to 34.

5.4 Optimization: Stepping Stones

The model and optimization framework were tested by planning several steps through a terrain with five ditches. At every step, the framework was designed to solve two optimization problems, one for stepping before and one for stepping after each ditch. A nonlinear constraint was imposed on the solver to account for this decision. After solving both optimization problems, the solution with the lowest cost is chosen. The optimization $m\hat{\theta}_1^{i+1}$ (Eqn. 11) and simulation $m\hat{\theta}_1^{i+1}$ (Eqn. 10) velocities are shown in Fig. 9. The velocities are informed by the control actions in the previous step which in turn are influenced by the terrain. The percent deviation from the nominal control is shown in Fig. 10. The planar biped took eight steps to traverse five ditches. The biped varied its step length to achieve clearance and proper foot placement in avoiding all the ditches, shown in [1]. The number of function evaluations ranged from 30 to 155.

6. DISCUSSION

A 5-link planar biped was reduced via PFL from a 10D to 2D state space. The system was reduced to 1D by taking the Poincaré sections at mid-stance. Previous approaches for orbital stabilization of legged robots were built around the limit cycle for control, but this control strategy was limited to small deviations from the fixed point. In contrast, data-driven approaches enable robust control capable of stabilizing a wider range of initial conditions. The system was simulated for various initial conditions to fit a GPC model capable of predicting whether a state-action pair would result in a feasible or infeasible step. The feasible data set was used to fit a GPR model capable of predicting the velocity at the next mid-stance step as a function of the current state-action pair, an approximation for the Poincaré map. The GPC and GPR models were set as constraints for a quadratic optimization.

We evaluated the optimization framework and model robustness through two trials, where the primary objectives were to track a reference velocity and step over obstacles. The velocity tracking was nearly flawless, with an average percent error of 0.21%. This performance demonstrated a simple scenario where the optimization strategy could be effective and efficient, with less than 34 function evaluations. To showcase the efficacy of our data-driven approach in solving a more complex task, we conducted the stepping stones optimization. Despite its increased complexity, we successfully identified optimal control inputs using few function evaluations. Our results, presented in Fig. 10, illustrate how the system modulates the control states away from nominal to step over the obstacles. In step number five, the biped required drastic deviation from the nominal controls to clear the large ditch. The optimization velocity, shown in blue in Fig. 9, decreased in step number six due to the control action taken in the previous step. Overall, the optimization velocity deviated less from the nominal than the simulation velocity possibly due to the optimization constraints, Eqns. 19 and 20. This could be remedied by refining the sampling region for the training data to obtain more feasible data to train the GPR. The lack of deviation from

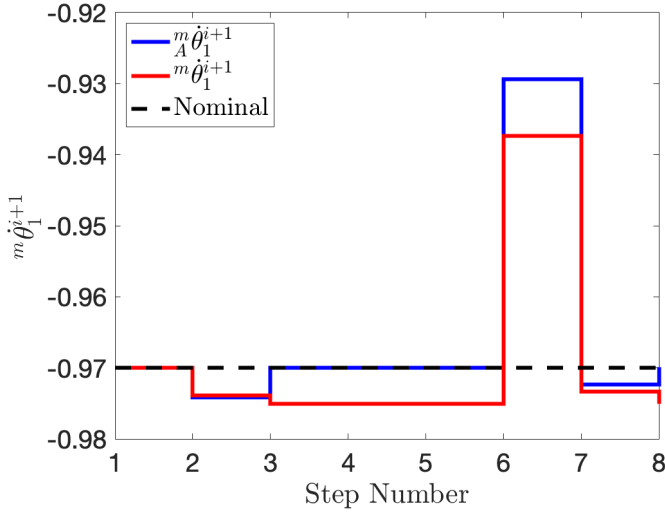


FIGURE 9: OPTIMIZATION AND SIMULATION VELOCITY AT EVERY STEP IN THE STEPPING STONES SIMULATION.

the nominal could cause issues in more complicated scenarios where more flexibility is required. For example, if the simulation contained sensor or process noise the optimization values would be impacted and deviate significantly. This could be corrected by training another GPR model to encompass the stochasticity in the system. An alternative is to use the Poincaré map model from the simulation and augment the model with a GPR noise model. The learned control policy with an augmented GPR model improved trajectory prediction. Despite the shortcomings, the optimization strategy proved capable of achieving high accuracy within a low number of function evaluations, 12 to 34 for velocity tracking and 30 to 155 for stepping stones. A similar optimization problem would be formulated on hardware; the expectation would be that a low number of function evaluations would equate to computational efficiency that would allow for real time control.

This method improves upon [13] which utilizes the same biped model, although it should be noted that there are differences in the simulations which do not make the comparisons one-to-one. The previous work approximated the region of validity with a support vector machine classifier and predicted feasibility (86.2%), infeasibility (95.8%), and mean accuracy (93.6%) compared to the GPC with feasibility (98.2%), infeasibility (100%), and mean accuracy (99.5%). The GPC accurately captures the region of the validity for this system. This work utilized GPR to approximate the Poincaré map and had a mean accuracy of 98.96% compared to a quadratic polynomial regression approximation with a mean accuracy of 95.61%. The GPR is able to capture the Poincaré map’s higher order terms, resulting in low error. Although a different velocity profile was tracked, this approach yielded low error, 0.21% compared to 10.6%, and resulted in a similar number of function evaluations, 12 to 34 compared to 15 to 40. The stepping stones simulation is difficult to compare as this approach was undertaken in a different environment with one more ditch. Despite this difference, both optimization methods remained close to the nominal velocity, except when taking longer steps. The max number of function evaluations was higher, 155

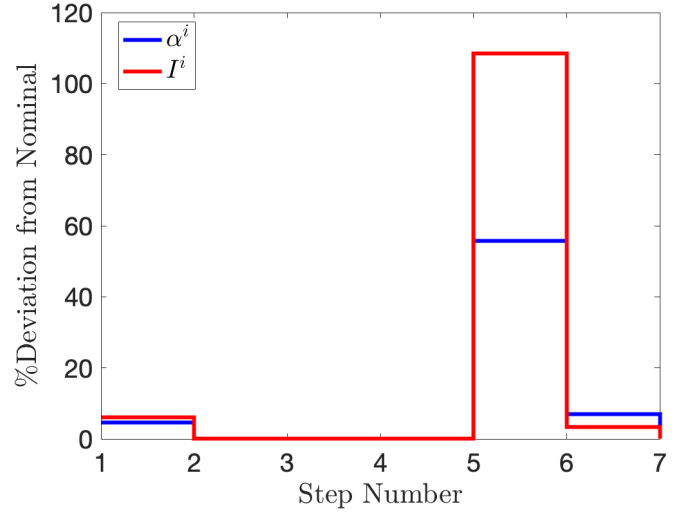


FIGURE 10: NOMINAL CONTROL PERCENT DEVIATION AT EVERY STEP IN THE STEPPING STONES OPTIMIZATION SIMULATION.

compared to 130.

7. CONCLUSIONS AND FUTURE WORK

We conclude that GPR and GPC are effective methods to model the Poincaré map and region of validity for bipedal walking. The implemented strategy can achieve computationally efficient and precise control in bipedal walking, thus proving a promising method to implement on hardware.

The optimization strategy can be further improved by augmenting the simulation with a GPR model to account for any model inaccuracies or stochasticity. This would further enhance the robustness and would facilitate the sim-to-real transfer. Eventually, this model will be evaluated on a humanoid with the goal of augmenting the model with hardware generated data, similar to the sim-to-real transfer method explored in [22].

ACKNOWLEDGMENTS

The work is funded by the US National Science Foundation through grant 2128568. Daniel Torres is funded by the US National Foundation through the Graduate Research Fellowship Program. Ernesto Hernandez Hinojosa is funded by the American Heart Association Predoctoral Fellowship.

REFERENCES

- [1] Torres, Daniel. “Stepping Stones.” <https://www.youtube.com/watch?v=p2TLU3Jzul8> (2023).
- [2] Hobbelen, D.G.E. and Wisse, M. “Limit cycle walking.” *Humanoid Robots Human-like Machines* (2007): pp. 277–294.
- [3] Pratt, Jerry, Carff, John, Drakunov, Sergey and Goswami, Ambarish. “Capture point: A step toward humanoid push recovery.” *2006 6th IEEE-RAS international conference on humanoid robots*: pp. 200–207. 2006. IEEE.
- [4] Xie, Zhaoming, Berseth, Glen, Clary, Patrick, Hurst, Jonathan and van de Panne, Michiel. “Feedback control for

- cassie with deep reinforcement learning.” *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*: pp. 1241–1246. 2018. IEEE.
- [5] Kuindersma, Scott, Deits, Robin, Fallon, Maurice, Valenzuela, Andrés, Dai, Hongkai, Permenter, Frank, Koolen, Twan, Marion, Pat and Tedrake, Russ. “Optimization-based locomotion planning, estimation, and control design for the atlas humanoid robot.” *Autonomous robots* Vol. 40 No. 3 (2016): pp. 429–455.
- [6] McGeer, T. “Passive dynamic biped catalogue.” *In Proc. of 2nd International Symposium on Experimental Robotics*: pp. 465–490. 1991.
- [7] McGeer, T. “Passive dynamic walking.” *The International Journal of Robotics Research* Vol. 9 No. 2 (1990): pp. 62–82.
- [8] McGeer, Tad. “Dynamics and control of bipedal locomotion.” *Journal of Theoretical Biology* Vol. 163 No. 3 (1993): pp. 277–314.
- [9] Wang, Austin Shih-Ping, Chen, William Wei-Lun and Lin, Pei-Chun. “Control of a 2-D bounding passive quadruped model with Poincaré map approximation and model predictive control.” *2016 International Conference on Advanced Robotics and Intelligent Systems (ARIS)*: pp. 1–6. 2016. IEEE.
- [10] Bhounsule, Pranav A, Kim, Myunghee and Alaeddini, Adel. “Approximation of the step-to-step dynamics enables computationally efficient and fast optimal control of legged robots.” *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Vol. 83990: p. V010T10A050. 2020. American Society of Mechanical Engineers.
- [11] Xiong, Xiaobin, Reher, Jenna and Ames, Aaron D. “Global position control on underactuated bipedal robots: Step-to-step dynamics approximation for step planning.” *2021 IEEE International Conference on Robotics and Automation (ICRA)*: pp. 2825–2831. 2021. IEEE.
- [12] Morimoto, Jun and Atkeson, Christopher G. “Nonparametric representation of an approximated poincaré map for learning biped locomotion.” *Autonomous robots* Vol. 27 (2009): pp. 131–144.
- [13] Hernandez-Hinojosa, Ernesto, Satici, Aykut and Bhounsule, Pranav A. “Optimal Control of a 5-Link Biped Using Quadratic Polynomial Model of Two-Point Boundary Value Problem.” *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, Vol. 85451: p. V08BT08A005. 2021. American Society of Mechanical Engineers.
- [14] Green, Kevin, Warila, John, Hatton, Ross L and Hurst, Jonathan. “Motion Planning for Agile Legged Locomotion using Failure Margin Constraints.” *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*: pp. 10350–10355. 2022. IEEE.
- [15] Grizzle, J.W., Abba, G. and Plestan, F. “Asymptotically stable walking for biped robots: Analysis via systems with impulse effects.” *IEEE Transactions on Automatic Control* Vol. 46 No. 1 (2001): pp. 51–64.
- [16] Westervelt, E.R. and Grizzle, J.W. “Design of asymptotically stable walking for a 5-link planar biped walker via optimization.” *Proceedings 2002 IEEE International Conference on Robotics and Automation (Cat. No.02CH37292)*, Vol. 3: pp. 3117–3122 vol.3. 2002. DOI 10.1109/ROBOT.2002.1013706.
- [17] Hereid, Ayonga, Cousineau, Eric A, Hubicki, Christian M and Ames, Aaron D. “3D dynamic walking with underactuated humanoid robots: A direct collocation framework for optimizing hybrid zero dynamics.” *2016 IEEE International Conference on Robotics and Automation (ICRA)*: pp. 1447–1454. 2016. IEEE.
- [18] Chevallereau, C., Grizzle, J.W. and Shih, C.L. “Asymptotically stable walking of a five-link underactuated 3-D bipedal robot.” *IEEE Transactions on Robotics* Vol. 25 No. 1 (2009): pp. 37–50.
- [19] Kuo, A.D. “Energetics of actively powered locomotion using the simplest walking model.” *Journal of Biomechanical Engineering* Vol. 124 (2002): pp. 113–120.
- [20] Strogatz, S.H. *Nonlinear dynamics and chaos*. Addison-Wesley Reading (1994).
- [21] Bhounsule, Pranav A, Hernandez Hinojosa, Ernesto and Alaeddini, Adel. “One-Step Deadbeat Control of a 5-Link Biped Using Data-Driven Nonlinear Approximation of the Step-to-Step Dynamics.” *Robotics* Vol. 9 (2020): p. 90.
- [22] Krause, Jeremy, Alaeddini, Adel and Bhounsule, Pranav A. “Gaussian Process Regression for Sim-to-Real Transfer of Hopping Gaits.” *ASME-International Design Engineering & Technical Conference*. 2023.