EECS 598 Final Project

Grant A Gibson, Jonathan B Michaux, Margaret Eva Mungai

I. Introduction

Bipedal legged locomotion research has progressed greatly over the past few decades. Many tasks such as walking, running, and jumping have been successfully implemented using legged robots. However, before any of these robots can achieve complex motions, they must first learn to balance. From the balance position, nearly all motions (including shut down) are realized. For this reason, the design of safe, balancing controllers is required in order to reduce failure occurrence (which can also lead to hardware deterioration).

The crux of our research is the ability to design controllers that guarantee safe dynamic bipedal locomotion.To prevent failure modes (i.e. falling), controllers must be able to reject unexpected external forces such as those which might arise from navigating rough terrain. This process is known as push recovery, and is essential for enabling bipedal robots to exhibit multiple modes of locomotion in the real world such as walking, running, and climbing stairs. In this report, we reproduce the capturability approximation results for the Linear Inverted Pendulum Model (LIPM) to achieve push recovery. More explicitly, we implement the outer approximation methods presented by Posa et al.[1].

II. Background

We begin by summarizing the background information necessary to understand how approximated viable-capture region are computed. We first describe the dynamics and configuration of the model being tested. Next we define capturability and its relation to stability and reachability analysis. We also define barrier functions and Sum-of-Squares programmings and the role they play in computing these methods approximately.

A. Dynamics & Model

The LIPM is a powerful model for legged locomotion because of its simplicity and applicability to various legged configurations. The continuous dynamics are linear and represent the Center of Mass (CoM) dynamics of a given robot. Note that the full-order dynamics models of legged systems are typically highly nonlinear and contain many degrees of freedom. This can make control development very difficult. By controlling a reduced order model (LIPM), which regulates CoM, simpler control policies can be derived and executed. The continuous dynamics are given (1) and a visual schematic is shown in Figure 1

$$\ddot{x}_{cm} = \frac{g}{z_{cm}}(x_{cm} + r_{foot}u_1) \tag{1}$$

where g is the gravity constant, r_{foot} is the width of the stance foot, L is the constant height of the center of mass,

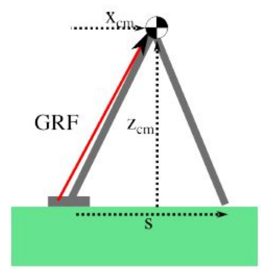

Fig. 1: Schematic for the Linear Inverted Pendulum Model. The location of the center of mass is indicated along with the direction of ground reaction forces. Once the x_{cm} state coordinate reaches the step length s, the discrete dynamics are applied, after which, the continuous dynamics continue in a periodic fashion.

and u_1 is the control input. x_{cm} and z_{cm} are horizontal and vertical CoM positions, respectively.

The discrete dynamics are applied instantaneously once x_{cm} reaches a given threshold, indicating the end of a step. We define these dynamics as well (2). Notice that the velocity of the state remains the same before (-) and after (+) impact. This is an assumption that can be relaxed in different variations of the LIPM.

$$r(x_-, s) = \begin{bmatrix} x_{cm-} - r_{step}s \\ \dot{x}_{cm-} \end{bmatrix} \tag{2}$$

We assume the following additional assumptions for our analysis of the LIPM: (i) The height of the center of mass remains constant, (ii) the ground reaction forces (GRF) of the model are constrained so that $\ddot{z}_{cm} = 0$, and (iii) the angular momentum is constant. Notice that from this formulation we see that the control input (or force applied at the hip) can only act horizontally in order to satisfy constraints. For our analysis we assume control implementation of a normalized bang-bang controller. For our analysis we assume $u_1 \in \{-1, 1\}$.

B. Capture Point & Capturability

To be effective, bipedal robots must be able to walk in varied environments in the presence of unmodeled disturbances. To ensure safety, it is therefore essential to measure how close a legged robot is to falling. Although existing methods such as zero moment point or Poincare map analysis have been used for measuring the stability of legged systems, they only apply to specific classes of controllers [2], [3]. Here, we focus on a controller-neutral approach using Capture Points and Capture Regions [4].

A Capture Point is defined as a point on the ground that allows the robot to bring itself to a complete stop [4]. A Capture Region is defined as the collection of all Capture

Points. The notion of Capture Point and Capture Region lead to a natural strategy for push recovery, as illustrated in Figure 2. If a Capture Point is located within the Base of Support, the robot is capable of recovering from a push without taking a step. If the robot's Base of Support intersects the Capture Region, then the robot can stop by taking a single step. If the Capture Region lies outside of the robot's workspace, the robot will fail to recover from a push in only one step.

Recent work from Koolen *et al* extends the notion of Capture Point and Capture Region to allow the robot to recover from a push by taking multiple steps [5]. Taking into account dynamics and actuator limits, N-step capturability is defined as the ability of a system to stop without falling by taking N or fewer steps. The N-step viable-capture basin is the region of state-space in which a controller can stabilize the system until it reaches a Capture Point by taking N or fewer steps. In this work, the viable-capture basin for the LIPM is the set of states where the robot is guaranteed to attain static stability. Note that a state is considered to be N-step capturable if a stepping event resets the state to an (N-1)-step capturable state. We leverage this recursive property in the methods described below.

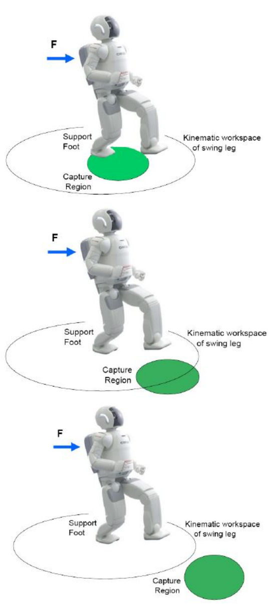

Fig. 2: (**Figure and caption reproduced from [4]**) When the Capture Region intersects the Base of Support, a humanoid can modulate its Center of Pressure to balance and does not need to take a step (top). When the Capture Region and Base of Support are disjoint, the humanoid must take a step to come to a stop (middle). If the Capture Region is outside the kinematic workspace of the swing foot, the humanoid cannot stop in one step (bottom figure).

C. Barrier Functions

Barrier functions have been used in numerous applications for safety verification as they don't require the direct computation of reachable sets [4], [6]. Given a set of unsafe states, χ_u, and a set of safe states, χ_s, one can use a barrier function, $V(t,x)$, to prove that χ_u is unreachable from any initial state in χ_s. For this application, χ_s are the states needed to get the LIPM to a viable-capture region, and χ_u are the states that will trigger a failure mode. Given the dynamic equation

$$\dot{x} = f(x) + g(x)u \tag{3}$$

the desired barrier function has to satisfy (4), (5) and (6)

$$V(t,x) > \rho(t) \qquad t \in [0,T] \quad \forall x \in \chi_u \tag{4}$$
$$V(t,x) \leq \rho(t) \qquad t \in [0,T] \quad \forall x \in \chi_s \tag{5}$$
$$\frac{d\rho}{dt} - \frac{\partial V(t,x)}{\partial x}(f+gu) - \frac{\partial V(t,x)}{\partial t} > 0 \tag{6}$$

where $V : \mathbb{R}_+ \times \mathbb{R} \to \mathbb{R}$, $\rho : \mathbb{R}_+ \to \mathbb{R}$, and T is the final time. If such a barrier function exists, its ρ-sublevel set gives the inner approximation of the viable-capture region.

To derive an outer approximation of the viable-capture region, we introduce a slack function $p : \mathbb{R}_+ \times \mathbb{R}^n \to R^m$ such that

$$p_i(x) \geq |\frac{\partial V(t,x)}{x} g_i(x)| \quad for\ i = 1,...m \tag{7}$$

Recall that we assume $u \in \{-1, 1\}$, therefore,

$$|\frac{\partial V(t,x)}{x} g(x)u| \ \leq \ |\frac{\partial V(t,x)}{x} g(x)| \ \leq \ p \tag{8}$$

Given p, and a set of goal states $\chi_{goal} \subset \chi_s$, we define a barrier function whose 0-superlevel set gives the outer approximation as follows:

$$V(t,x) \leq 0 \qquad t \in [0,T] \quad \forall x \in \chi_u \tag{9}$$
$$V(t,x) > 0 \qquad t \in [0,T] \quad \forall x \in \chi_s \tag{10}$$
$$-\frac{\partial V(t,x)}{\partial x}(f(x) - \mathbf{1}^\intercal p - \frac{\partial V(t,x)}{\partial t} > 0 \tag{11}$$

We note that if $x \in \chi_{goal}$, then $V(T,x) > 0$. Sums-of-Squares programming can be used to calculate the barrier function.In this paper, we focus solely on the outer approximation of the viable-capture region.

D. Sum-of-Squares (SOS)

The Sum-of-Squares (SOS) problem is a an optimization with a specially defined structure (12). The problem has a cost function (that is linear in the coefficients of the unknown parameters) and all inequalities are written as polynomials which can be represented as a sum of squares. Explicit conditions for when certain polynomials can be converted to SOS are given in [7].

$$\begin{aligned} &\min\ c^T x \\ &\quad \textbf{subject to} \\ &\quad f(x) \text{ is SOS} \end{aligned} \tag{12}$$

This type of structured optimization program can then be converted to a Semi-Definite Program (SDP) for which minimizations are more easily solved. This conversion utilizes the S-procedure to equate an individual semi-algebraic inequality with several linear non-negative inequality constraints. The S-procedure method for the general problem formulation (12) is shown in (13) when given $(g(x) \geq 0,\ h(x) = 0)$. A rigorous example is explained in [7] (Example 5.3), and this procedure is ultimately utilized when formulating the outer approximation optimization problems.

$$\begin{aligned} \sigma_1(x)f(x) - \sigma_2(x)g(x) - q(x)h(x) &\geq 0 \\ \sigma_1(x) - 1 &\geq 0 \\ \sigma_2(x) &\geq 0 \end{aligned} \tag{13}$$

Where $\sigma_1, \sigma_2 \geq 0$.

III. Outer Approximation for Capturability

Here we present the semidefinite programs (SDPs) that are used to compute the outer approximation viable capture regions for 0-step (balancing) and N-step. The goal for the 0-step is to approximate all the states that a controller can stabilize at the origin without the LIPM taking a step. Similarly, the goal for the N-step is to approximate, over finite time, all the states that can result in a $(N-1)$ step viable capture region. In order to produce a *tight* outer approximation, we minimize a function $W : \mathbb{R}^n \to \mathbb{R}$ defined as

$$W(x) \geq 0 \quad \forall x \in \chi_s \tag{14}$$

$$W(x) \geq 1 + V(0,x) \quad \forall x \in \chi_s \tag{15}$$

We note that the tightness of the outer approximation will depend on the polynomial degrees chosen for W and V. To ensure that the set of feasible points has a non-empty interior, a new constraint (16) is introduced

$$x^{\mathsf{T}} x \leq R \tag{16}$$

where R is a slack variable. With this constraint in place, the failed states are outside the ball of radius R [8].

A. 0-step Formulation

The SDP optimization problem posed for the 0-step viable capture region is based on shown in (O1).

$$\begin{aligned} \min_{V,W,p,\sigma_R} \quad & \int_{B_R} W dx \qquad \text{(O1)} \\ \text{s.t.} \quad & -\frac{\partial V}{\partial x} f - \mathbf{1}^T p - \sigma_R(R^2 - x^T x) \text{ is SOS}, \\ & V|_{x=0} > 0, \\ & p_i - \frac{\partial V}{\partial x} g_i - \sigma_{p,i}(R^2 - x^T x) \text{ is SOS for } i = 1, \ldots, m, \\ & p_i + \frac{\partial V}{\partial x} g_i - \sigma_{p,i}(R^2 - x^T x) \text{ is SOS for } i = 1, \ldots, m, \\ & W \text{ is SOS}, \\ & W - V - 1 \text{ is SOS}, \end{aligned}$$

where $V : \mathbb{R}^n \to \mathbb{R}$, $p : \mathbb{R}^n \to \mathbb{R}^m$, and the σ's are multipliers obtained from the S-procedure discussed in Section II-D. The constraints are based on (7)-(11). The optimization variables are V, W, p, σ_R ; R, g, and f are defined prior to solving the SDP.

B. N-step Formulation

In the N-step formulation we include a condition that ensures that the N-step region leads to a $(N-1)$ viable capture region. This is achieved by first defining a set

$$\begin{aligned} H := \{(x,s,\Lambda) :\, & s \in [-1,1], h(x,s,\Lambda) = 0, \ldots \\ & V_{N-1}(0, r(x,s,\Lambda)) \geq 0\} \end{aligned} \tag{17}$$

where Λ is the impulse, $r(x,s,\Lambda)$ is the reset map, and $h(x,s,\Lambda) = 0$ is an implicit constraint that has to be satisfied. We then require (18) to be met.

$$V_N(T,x) \geq 0 \quad \forall (x,s,\Lambda) \in H \tag{18}$$

The N-step formulation is described in (O2)

$$\begin{aligned} \min_{V_N,W,p,q_h,\sigma_*} \quad & \int_{B_R} W dx \qquad \text{(O2)} \\ \text{s.t.} \quad & -\frac{\partial V}{\partial x} f - \frac{\partial V}{\partial t} - \mathbf{1}^T p - \ldots \\ & \quad \sigma_T(Tt - t^2) - \sigma_R(R^2 - x^T x) \text{ is SOS}, \\ & V_N(T,x) - \sigma_v V_{N-1}(0, r(x,s,\Lambda)) - \sigma_s(1 - s^2) - q_h h \\ & \quad \text{is SOS}, \\ & p_i - \frac{\partial V_N}{\partial x} g_i - \sigma_{Rp,i}(R^2 - x^T x) - \sigma_{Tp,i}(Tt - t^2) \ldots \\ & \quad \text{is SOS for } i = 1, \ldots, m, \\ & p_i + \frac{\partial V_N}{\partial x} g_i - \sigma_{Rn,i}(R^2 - x^T x) - \sigma_{Tn,i}(Tt - t^2) \ldots \\ & \quad \text{is SOS for } i = 1, \ldots, m, \\ & W \text{ is SOS}, \\ & W - V_N(0,x) - 1 \text{ is SOS}, \\ & \sigma_R, \sigma_T, \sigma_V, \sigma_s, \sigma_{Rp,i}, \sigma_{Rn,i}, \sigma_{Tp,i}, \sigma_{Tn,i} \text{ are SOS}, \end{aligned}$$

where the σ's and q_h are multipliers. The optimization variables are V_N, W, p, q_h, σ_*; R, T, f, g, h and V_{N-1} are defined prior to solving the SDP.

IV. Results: Outer Approximation

In this section, we reproduce outer approximations of the 0-step and 1-step viable capture regions for the LIPM. These results are shown in Figure 3 and the explicit solution polynomials are given in (19).

$$\begin{aligned} V_{0-step} &= -0.6607x_1^2 - 4.2622x_1x_2 - 6.8419x_2^2 \\ V_{n-step} &= -0.0551x_1^2 - 0.3836x_1x_2 - 0.6157x_2^2 \end{aligned} \tag{19}$$

Here the colored regions correspond to initial state values which guarantee capturability. There are many parameters needed in order to solve the defined optimization problems (O1) and (O2); their values are listed in Table I. Although the paper uses different solvers we still managed to obtain results using SOSTools for SOS problem formulation and Sedumi for solving the subsequent Semi-Definite Programming problem (Sedumi) [9],[10].

An important clarification must be made with regards to solving the N-step optimization problem (O2). V_{N-1} represents the barrier function solution for the $N-1$ capture

TABLE I: Optimization parameter values used to derive 0-step and 1-step capturability results for the LIPM.

Quantity	Value	Polynomial	Degree
T	0.3 s	V	2
r_{step}	0.7 m	W	4
r_{foot}	0.05 m	p	4
$\bar{z}_{cm}$	1 m	σ_*	1 (scalars)
R	2	-	-

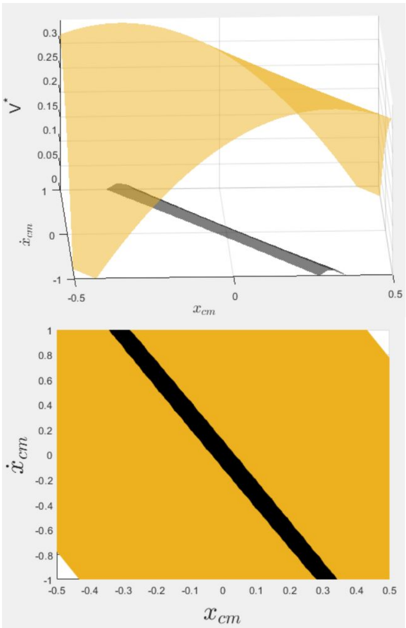

Fig. 3: Outer Approximations for the LIPM. The top plot displays the three-dimensional barrier function solution for the 0-step (black) and 1-step (orange) viable-capture regions. The bottom plot is a top-down view of these functions.

region and so we initialize this value with the solution to our 0-step solution. We also note that for the standard LIPM problem there is no impact, therefore, the q_h and h terms were omitted.

V. Discussion and Future Work

This report describes our approach for reproducing the results in [1]. In particular, we computed outer approximations of the 0-step and 1-step viable-capture regions for a planar LIPM. These outer approximations delineate regions of state-space where the LIPM is guaranteed to balance in 0 or 1-step, respectively. The original problem was formulated as a Sums-of-Squares program and then solved after converting it to a SDP.

We were able to attain similar results to the original paper despite minor differences in implementation. For example, the original work made use of Spotless and MOSEK for formulating and solving the optimization problems. Here, we used SOSTools and Sedumi. We also explored different parameter values. Notably, we explored various values for the degrees of the polynomials V and W. We observed that the size of the feasible region had an inverse relation with the degree of the polynomial (data not shown).

Another difference in our implementation involved the computation of the barrier function V_{N-1}. Since V_{N-1} was not described in the paper, we simply used the previous outer approximation as an initialization. We note that it would also be possible to initialize the optimization using either the analytical derivation of the 0-step region or the inner approximation.

In the future we would first like to solve the inner approximation for the LIPM. We anticipate that calculating the outer and inner approximations for the LIPM viable-capture regions is an excellent starting point for analyzing the stability of various higher-order models. In particular, we would like to extend this method beyond the simple models considered in [1] to bipedal models such as the compass-gait (2-link) and rabbit (5-link) [11]. It is predicted that the complexity of these models might be within the scale of current state-of-the art SOS programs. The results, therefore, will ultimately prove insightful when analyzing higher order legged systems such as Cassie or Digit [12].

Appendix I
Team Contributions

Each person contributed equally to this report.

Appendix II
Code

A github repository of our code can be found at https://github.com/grantgib/SoS_LIP_Model. The repository contains all of our code for optimizations and plot generations.

References

[1] M. Posa, T. Koolen, and R. Tedrake, "Balancing and step recovery capturability via sums-of-squares optimization." in *Robotics: Science and Systems*. Cambridge, MA, 2017, pp. 12–16.

[2] K. Masani, A. H. Vette, and M. R. Popovic, "Controlling balance during quiet standing: Proportional and derivative controller generates preceding motor command to body sway position observed in experiments," *Gait Posture*, vol. 23, no. 2, pp. 164 – 172, 2006. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0966636205000329

[3] D. Hobbelen and M. Wisse, "Controlling the walking speed in limit cycle walking," *The International Journal of Robotics Research*, vol. 27, no. 9, pp. 989–1005, 2008. [Online]. Available: https://doi.org/10.1177/0278364908095005

[4] J. Pratt, J. Carff, S. Drakunov, and A. Goswami, "Capture point: A step toward humanoid push recovery," in *2006 6th IEEE-RAS international conference on humanoid robots*. IEEE, 2006, pp. 200–207.

[5] T. Koolen, T. De Boer, J. Rebula, A. Goswami, and J. Pratt, "Capturability-based analysis and control of legged locomotion, part 1: Theory and application to three simple gait models," *The international journal of robotics research*, vol. 31, no. 9, pp. 1094–1113, 2012.

[6] M. M. Tobenkin, I. R. Manchester, and R. Tedrake, "Invariant funnels around trajectories using sum-of-squares programming," *IFAC Proceedings Volumes*, vol. 44, no. 1, pp. 9218–9223, 2011.

[7] P. A. Parrilo, "Semidefinite programming relaxations for semialgebraic problems," *Mathematical programming*, vol. 96, no. 2, pp. 293–320, 2003.

[8] C. Josz and D. Henrion, "Strong duality in lasserre's hierarchy for polynomial optimization," *Optimization Letters*, vol. 10, no. 1, pp. 3–10, 2016.

[9] A. Papachristodoulou, J. Anderson, G. Valmorbida, S. Prajna, P. Seiler, and P. Parrilo, "Sostools version 3.00 sum of squares optimization toolbox for matlab," *arXiv preprint arXiv:1310.4716*, 2013.

[10] Y. Labit, D. Peaucelle, and D. Henrion, "Sedumi interface 1.02: a tool for solving lmi problems with sedumi," in *Proceedings. IEEE International Symposium on Computer Aided Control System Design*. IEEE, 2002, pp. 272–277.

[11] E. R. Westervelt, J. W. Grizzle, C. Chevallereau, J. H. Choi, and B. Morris, *Feedback control of dynamic bipedal robot locomotion*. CRC press, 2018.

[12] A. Robotics, "Robots," accessed: April 2020. [Online]. Available: https://www.agilityrobotics.com/robots

[13] S. Prajna and A. Jadbabaie, "Safety verification of hybrid systems using barrier certificates," in *International Workshop on Hybrid Systems: Computation and Control*. Springer, 2004, pp. 477–492.