The Thesis Committee for Ye Zhao
Certifies that this is the approved version of the following thesis:

Phase Space Planning for Robust Locomotion

APPROVED BY
SUPERVISING COMMITTEE:

Supervisor:

Luis Sentis

Ashish Deshpande

Phase Space Planning for Robust Locomotion

by

Ye Zhao, B.E.

THESIS

Presented to the Faculty of the Graduate School of

The University of Texas at Austin

in Partial Fulfillment

of the Requirements

for the Degree of

MASTER OF SCIENCE IN ENGINEERING

THE UNIVERSITY OF TEXAS AT AUSTIN

August 2013

Dedicated to my parents and girlfriend.

Acknowledgments

I would like to thank numerous people who make this thesis possible.

Foremost, I would like to express my sincere gratitude to my research advisor, Prof. Luis Sentis, for his endless support and guidance. He always instructs me to carry out research and achieve solid results step by step. Many ideas are inspired from our deep-going discussions. More importantly, his strictness and methodology for doing research influence me profoundly and will be a life fortune for my future career. I would also thank Prof. Ashish Deshpande for serving as my committee member. His advices are really precious. Also, appreciate Prof. Benito Fernandez's suggestions and guidance.

Next, I would like to acknowledge my lab members. Donghyun Kim, Kwan Suk Kim have been a crucial source of research brainstorming. The boom system constructed by Alan Kwok helps a lot for experiments. Nick Paine gives a lot of research suggestions. Appreciate the whole body control instruction from Gwen Johnson and Chien-Liang Fok. I am grateful for the help from Kenan Isik, Cory Crean and Mike Slovich.

Lastly, I would like to thank my parents for their love, patience and support. Without them I won't be able to imagine who I would be now. Also thanks to my girlfriend for her trust and support. Family is always my power in doing anything.

Phase Space Planning for Robust Locomotion

Ye Zhao, M.S.E.
The University of Texas at Austin, 2013

Supervisor: Luis Sentis

Maneuvering through 3D structures nimbly is pivotal to the advancement of legged locomotion. However, few methods have been developed that can generate 3D gaits in those terrains and fewer if none can be generalized to control dynamic maneuvers. In this thesis, foot placement planning for dynamic locomotion traversing irregular terrains is explored in three dimensional space. Given boundary values of the center of mass' apexes during the gait, sagittal and lateral Phase Plane trajectories are predicted based on multi-contact and inverted pendulum dynamics. To deal with the nonlinear dynamics of the contact motions and their dimensionality, we plan a geometric surface of motion beforehand and rely on numerical integration to solve the models. In particular, we combine multi-contact and prismatic inverted pendulum models to resolve feet transitions between steps, allowing to produce trajectory patterns similar to those observed in human locomotion. Our contributions lay in the following points: (1) the introduction of non planar surfaces to characterize

the center of mass' geometric behavior; (2) an automatic gait planner that simultaneously resolves sagittal and lateral feet placements; (3) the introduction of multi-contact dynamics to smoothly transition between steps in the rough terrains.

Data driven methods are powerful approaches for modeling in absence of physical models. These methods rely on experimental data for trajectory regression and prediction. Here, we use regression tools to plan dynamic locomotion in the Phase Space of the robot's center of mass and we develop nonlinear controllers to accomplish the desired plans with accuracy and robustness. In real robotic systems, sensor noise, simplified models and external disturbances contribute to dramatic deviations of the actual closed loop dynamics with respect to the desired ones. Moreover, coming up with dynamic locomotion plans for bipedal robots and in all terrains is an unsolved problem. To tackle these challenges we propose here two robust mechanisms: support vector regression for data driven model fitting and contact planning, and trajectory based sliding mode control for accuracy and robustness. First, support vector regression is utilized to learn the data set obtained through numerical simulations, providing an analytical solution to the nonlinear locomotion dynamics. To approximate typical Phase Plane behaviors that contain infinite slopes and loops, we propose to use implicit fitting functions for the regression. Compared to mainstream explicit fitting methods, our regression method has several key advantages: 1) it models high dimensional Phase Space states by a single unified implicit function; 2) it avoids trajectory over-fitting; 3) it guar-

antees robustness to noisy data. Finally, based on our regression models, we develop contact switching plans and robust controllers that guarantee convergence to the desired trajectories. Overall, our methods are more robust and capable of learning complex trajectories than traditional regression methods and can be easily utilized to develop trajectory based robust controllers for locomotion. Various case studies are analyzed to validate the effectiveness of our methods including single and multi step planning in a numerical simulation and swing foot trajectory control on our Hume bipedal robot.

Table of Contents

List of Figures

Chapter 1

Introduction

How is it that many legged animals are capable to nimbly maneuver on 3D surfaces but humanoid robots can only slowly walk on them? To tackle this deficiency, we aim at developing new models characterizing 3D legged dynamics and designing methods to find 3D feet placements that achieve the desired gait regimes. To do so, this thesis presents a new 3D agile motion planner capable to maneuver in irregular terrains and in a natural manner. Additionally, a data driven method is utilized to derive Phase Space implicit regression trajectories of motion planning. Based on this implicit trajectory, sliding mode controller is developed to gain Phase Space robustness when external disturbance appears. As such, this planning and control strategy is aimed to control semi-autonomous legged robots in realistic outdoor environments.

1.1 The State of The Art: Locomotion Planning

Motion planning in robotics has been a focus of attention since obstacle-free mobility became a research topic. We will not attempt here to cover the full scope of motion planning, but a good reference can be found at [34]. Early works on motion planning focused on graph based search, but didn't scale well

Figure 1.1: A conceptual biped robot with solar panel in grand canyon

.

to highly articulated robotic systems. To solve this issue, randomized search methods were proposed, such as rapidly-exploring randomized trees [36], or probabilistic roadmaps [4], among others. However, these methods only consider the robot's configuration space, but not its dynamic behavior. As a result, extensions to kinodynamic planning where later proposed, e.g. [37, 69]. One problem on the early methods is that motor control policies were separated from the motion plans preventing robustness and realtime responsiveness. To address this issue, motion planners were proposed in [71] that incorporate the feedback controller into the trajectory design. Disturbances and uncer-

tainty of the motion plans have been recently studied for realtime control in [40]. Synthesis of complex contact behaviors has been thoroughly studied by [46], thought not focusing in dynamic bipedal locomotion. Learning switching policies has been addressed in [53] in the context of planar locomotion and using model reduction, and more recently in [74] using full joint dynamics but limited to planar gaits.

Legged robot locomotion has attracted extensive attention in recent years and many humanoids have been developed to achieve agile and efficient locomotion behaviors [9, 15, 76]. Here we propose a new generation of conceptual solar bipedal robot in Fig. 1.1. This sort of robot aims at the capabilities of traversing rugged mountains and grand canyons for long durations. In literature, many studies attempt to advance the state-of-the-art in rough terrain locomotion [5, 31, 49, 51, 52] and many works focus on analyzing dynamic stability [14, 15, 20, 24, 26, 41]. In particular, current bipedal methods using linearized assumptions cannot achieve human like speeds in rough terrains due to the simplicity of the models, and methods relying on stability analysis require the analysis of periodic trajectories which do not apply to rough terrains. The Capture Point method described in [51] represents a powerful framework to plan feet placements. However, compared to our methods described here it relies on linearized model. In contrast, our methods are competitive to non-flat 3D rough terrains because we do not utilize linearized models. Grizzle et al., [76] uses competitive to design the walking gait. By selecting Bezier polynomials, they obtain an exponentially stable orbit for the closed-loop sys-

tem instead of the traditional optimal or approximately optimal open-loop trajectory.

1.2 3D Foot Placement Planning on Rough Terrain

We accomplish the 3D rough terrain capabilities by doing the following: (1) we develop prismatic inverted pendulum dynamics to describe the sagittal and lateral single contact behaviors, (2) we develop multi-contact models to describe the dynamics and internal forces of dual contact phases, (3) we introduce non-planar center of mass surfaces of motion to reduce the dimensionality of the model dynamics, (4) given desired feet step locations in the sagittal plane and desired apex sagittal velocities of the steps, we use numerical integration to solve sagittal feet phase placements, (5) to smoother peak velocities, we incorporate multi-contact phases and solve for the corresponding dynamics given surface friction constraints, (6) we then extract time profiles of the generated steps and use them to search lateral feet placements that keep the gait within velocity bounds, (7) we extract time profiles of the center of mass and feet trajectories for verification and control, (8) we apply our algorithm to the terrains with irregular profiles to demonstrate the validity of our work.

One of the main characteristics of the proposed study is its generalizing principles, such as combining various contact models, relying on numerical methods, solving for feet placements in the phase plane, and maintaining center of mass movement within velocity bounds. We show the potential of our

techniques in the generation of gait by maneuvering nimbly in a terrain with strong height variations using a biped visualization environment and comparing it to the performance of a human walking. To validate the applicability of our algorithm, our planner is simulated on three different challenging terrain sets. Also, we have recently shown extensions of some of our methods to other gaits such as walking on vertical surfaces [61] or producing brachiation gaits [63]. Similar ideas could be used for generating gaits on quadruped robots. For instance in [25] it is shown that controlling internal tensions during rough terrain walking allows a quadruped to prevent slippage over an inclined surface.

1.3 Phase Space Implicit Regression

A major goal in locomotion is to develop models and control policies that can simplify the maneuvering of robots in all terrains and provide robustness to disturbances. Data driven models [75] are an attractive approach since they can capture the robot's behavior in the complex terrains [16], robustness to the uncertainty [19] and the freedom from sophisticated modeling [53]. Simple regression models such as polynomial fitting are often used to learn the robot models [76], but suffer from conditioning problems, trajectory over-fitting and cannot capture the full complexity of the dynamic behaviors. Among various data driven fitting methods, support vector regression (SVR) is one of the most robust approaches and will be leveraged here for locomotion planning and control. SVR is derived from support vector machines, which was first

developed to solve classification problems [73], and then extended to regression problems [66].

The basic idea of SVRs is to map data sets onto a higher dimensional space via nonlinear mapping. The SVRs recently gains increasing attentions in the context of motion planning and control. For instance, it is utilized to classify the obstacles and then generate a collision-free path in 2D and 3D case [44]. However, this planning is only carried out in configuration space while the dynamics [68] is ignored. In our study, the SVRs is implemented in Phase Space to consider the system dynamics as well. We employ SVRs to robustly plan biped locomotion behaviors given simulation based data sets in Phase Space. The output of our training model is defined as an implicit surface function that allows to represent complex trajectories that we derive via the numerical simulations. Compared to purely kinematic planning, dynamic locomotion calls for more sophisticated plans such as our trajectory generation approach in the Phase Space of the robot's center of mass. The Phase Space is an ideal coordinate system to study stability and robustness of control systems. As we will show, it allows us to derive contact transition policies for switching locomotion dynamics of the hybrid system.

1.4 Robust Control

For biped locomotion, there often exists external force and disturbance [17, 19]. Sliding mode control [72] is a robust control strategy which benefits disturbance rejection. The robot dynamics can be represented by a sliding

manifold [77] in Phase Space. Then original system states can be reduced to one parameter describing the motion along the manifold and the other one describing the motion orthogonal to the manifold. Our study employs robust control to reject disturbance. In experimental part, we implement robust control to make robot feet take a step.

In this thesis, we develop feedforward controllers based on the planned trajectories, and robust feedback controllers to tolerate uncertainties. In particular, sliding mode control will be used to guarantee convergence when controlling the feet trajectories. Given our long term focus on acceleration based controllers (see Whole-Body Compliant Control [59]), the State Space motion plans derived here can be easily leveraged to produce robust feedforward and feedback control laws that can be directly sent to the motor control layer. The validity of our results is demonstrated in various simulations and in an experimental setup involving swing foot trajectory control of our bipedal robot Hume.

1.5 Contribution of This Thesis

Our contributions originate from the 3D gait planner and the data driven modeling and control.

First, the contributions of the planner lay in the following points: (1) the introduction of non planar surfaces to characterize the center of mass' geometric behavior; (2) an automatic gait planner that simultaneously resolves sagittal and lateral feet placements; (3) the introduction of multi-contact phase

to smoothly transition between steps in the rough terrains. Overall, the main contributions is to propose a framework to find 3D feet placement transitions at variable speeds in arbitrary rough terrains.

For the data driven modeling and control, the contributions come from: 1) Developing robust controllers that leverage the hybrid trajectory plans learnt from the regression process. 2) Validating the planners and robust controls in simulations and in an experimental setup involving our Hume bipedal robot.

Chapter 2

Inverted Pendulum Dynamics

In this chapter, we introduce the 3D inverted pendulum dynamics. Sagittal and lateral single contact dynamics are formulated respectively. In our study, numerical integration is proposed to deal with dynamics nonlinearities.

2.1 Prismatic Inverted Pendulum Dynamics

When considering 3D locomotion, sagittal and lateral single contact behaviors are coupled together making the foot placement generation a difficult task. However, with the assumption that the center of mass moves on a piecewise linear 3D surface, the sagittal and lateral dynamics become decoupled and therefore can be independently solved.

Using dynamic balance of moments, the difference between the moments acting on the contact foot and the net inertial and gravitational moments, is zero. Therefore, for the single contact scenario moment balance can be written as

$$p_{cop_k} \times f_{r_k} = p_{com} \times \Big(f_{com} + M\, g \Big) + m_{com}. \tag{2.1}$$

where, k is the limb in contact with the ground, p_{cop_k} is the limb's center of pressure (CoP) point, f_{r_k} is the three dimensional vector of reaction forces,

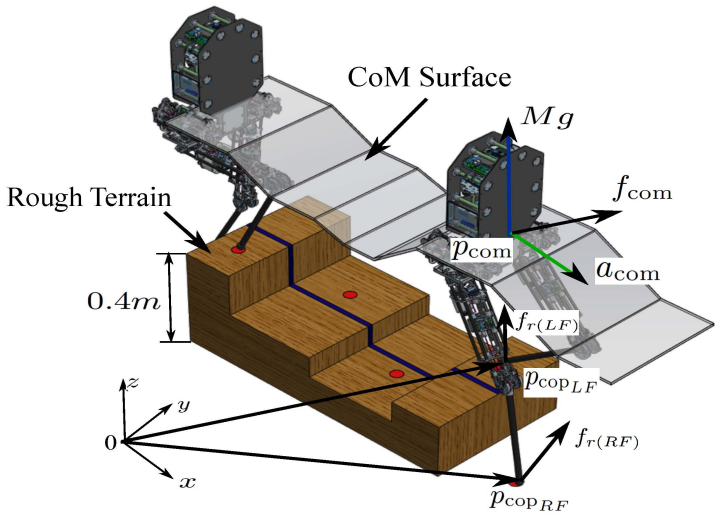


Figure 2.1: **3D schematic diagram of walking profile.** The center of mass geometric surface and the feet locations on the sagittal plane are provided by the gait designer and can take arbitrary forms as long as they are kinematically feasible. The center of mass position is p_{com}, the center of pressure (CoP) positions of the right and left feet are $p_{cops(LF)}$ and $p_{cops(RF)}$, CoM accelerations are a_{com}, and reaction forces are $f_{r(LF)}$ and $f_{r(RF)}$.

.

f_{com} and m_{com} are the three dimensional vectors of center of mass inertial forces and moments respectively, and g corresponds to the gravity field. The above equation is vectorial and determines three orthogonal moments. Force equilibrium can be formulated as $f_{r_k} = f_{com} + M\,g$, which allows to rewrite Equation (2.1) as

$$\left(p_{com} - p_{cop_k}\right) \times f_{r_k} = -m_{com}. \tag{2.2}$$

For our prismatic inverted pendulum model we assume single point mass [29], [28] and therefore inertial moments about the center of mass can be ignored, i.e. $m_{com} = 0$. As such, the above equation can be rewritten in vectorial form

as

$$\begin{pmatrix} 0 & -f_{r_{[kz]}} & f_{r_{[ky]}} \\ f_{r_{[kz]}} & 0 & -f_{r_{[kx]}} \\ -f_{r_{[ky]}} & f_{r_{[kx]}} & 0 \end{pmatrix} \begin{pmatrix} p_{com_x} - p_{cop_x} \\ p_{com_y} - p_{cop_y} \\ p_{com_z} - p_{cop_z} \end{pmatrix} = 0. \tag{2.3}$$

Using the equalities $f_{r_{[kx]}} = Ma_{com_x}$, $f_{r_{[ky]}} = Ma_{com_y}$ and $f_{r_{[kz]}} = M(a_{com_z} + g)$, we can decompose the above equation into the following three ones

$$a_{com_x} = \frac{\Big(p_{com_x} - p_{cop_x}\Big)\Big(a_{com_z} + g\Big)}{p_{com_z} - p_{cop_z}}, \tag{2.4}$$

$$a_{com_y} = \frac{\Big(p_{com_y} - p_{cop_y}\Big) \cdot a_{com_x}}{p_{com_x} - p_{cop_x}}, \tag{2.5}$$

$$a_{com_z} = \frac{\Big(p_{com_z} - p_{cop_z}\Big) \cdot a_{com_y}}{p_{com_y} - p_{cop_y}} - g. \tag{2.6}$$

where $a_{com[.]}$ represents the center of mass acceleration. We will use these three equations to formulate sagittal and lateral dynamic behavior.

2.2 Center of Mass Geometric Surface

Equations (2.4) to (2.6) are not only nonlinear but also multivariate and therefore they pose a problem to solve them. To deal with this difficulty we first reduce the dimensionality of the equations by planning a geometric surface of center of mass behavior beforehand. In Figure 2.1 we depict an example of a handmade surface. In this chapter we don't explore the making of the surfaces and assume they are giving to us. In this case, the surface is piecewise linear and it approximately follows the contour of the terrain. Our surface can

be expressed as

$$p_{com_z} = \begin{cases} a_1\, p_{com_x} + b_1, & p_{com} \in \mathbb{P}_1 \\ a_2\, p_{com_x} + b_2, & p_{com} \in \mathbb{P}_2 \\ \vdots & \\ a_N\, p_{com_x} + b_N, & p_{com} \in \mathbb{P}_N \end{cases} \tag{2.7}$$

where, $\mathbb{P}_i$ represents the path of the CoM over surface segment i. Moreover, the acceleration profile can be extracted by differentiating twice the above piecewise equation, i.e.

$$\text{if } p_{com_z} = a_i\, p_{com_x} + b_i, \text{ then } a_{com_z} = a_i\, a_{com_x}. \tag{2.8}$$

Let us first solve the sagittal inverted pendulum dynamics. Plugging the position and acceleration dependencies described in (2.8) into Equation (2.4) we get

$$a_{com_x} = \frac{\Big(p_{com_x} - p_{cop_x}\Big)\Big(a_i\, a_{com_x} + g\Big)}{a_i\, p_{com_x} + b_i - p_{cop_z}}. \tag{2.9}$$

and since a_{com_x} appears both on the left and right hand sides, we can rewrite the equation as

$$a_{com_x} = \frac{\Big(p_{com_x} - p_{cop_x}\Big) \cdot g}{\Big(a_i\, p_{cop_x} + b_i - p_{cop_z}\Big)}. \tag{2.10}$$

The above equation represents an inverted pendulum of variable height that tracks the desired surface. Therefore we call it the prismatic inverted pendulum model and we use it to describe single contact behaviors. Notice that by defining the center of mass surface in Figure 2.1, our sagittal model has

now become an ordinary differential equation that can be easily solved via numerical integration.

Let us now focus on the lateral single contact dynamics. Plugging the position and acceleration dependencies of Equation (2.8) into (2.5), we get

$$a_{com_y} = \frac{\Big(p_{com_y} - p_{cop_y}\Big) \cdot a_{com_z}}{\Big(p_{com_z} - a_i\, p_{cop_x} - b_i\Big)}\,. \tag{2.11}$$

Equation (2.6) can be rewritten by reorganizing terms as

$$a_{com_y} = \frac{\Big(p_{com_y} - p_{cop_y}\Big)}{\Big(p_{com_z} - p_{cop_z}\Big)}\Big(a_{com_z} + g\Big). \tag{2.12}$$

Expressing a_{com_z} from (2.11) in terms of a_{com_y} and plugging it above we get

$$a_{com_y} = \frac{\Big(p_{com_z} - a_i\, p_{cop_x} - b_i\Big)a_{com_y}}{\Big(p_{com_z} - p_{cop_z}\Big)} + \frac{\Big(p_{com_y} - p_{cop_y}\Big)}{\Big(p_{com_z} - p_{cop_z}\Big)}g. \tag{2.13}$$

Isolating a_{com_y} from above, the term a_{com_z} disappears and the above equation can be written as

$$a_{com_y} = \frac{\Big(p_{com_y} - p_{cop_y}\Big) \cdot g}{\Big(a_i\, p_{cop_x} + b_i - p_{cop_z}\Big)}. \tag{2.14}$$

This result is important as it represents lateral single contact dynamics as an ordinary differential equation too, and therefore independent of the dynamics of the sagittal dynamics.

2.3 Numerical Integration

Although we have found ordinary differential expressions that are decoupled for the sagittal and lateral planes, Equations (2.10) and (2.14) are usually

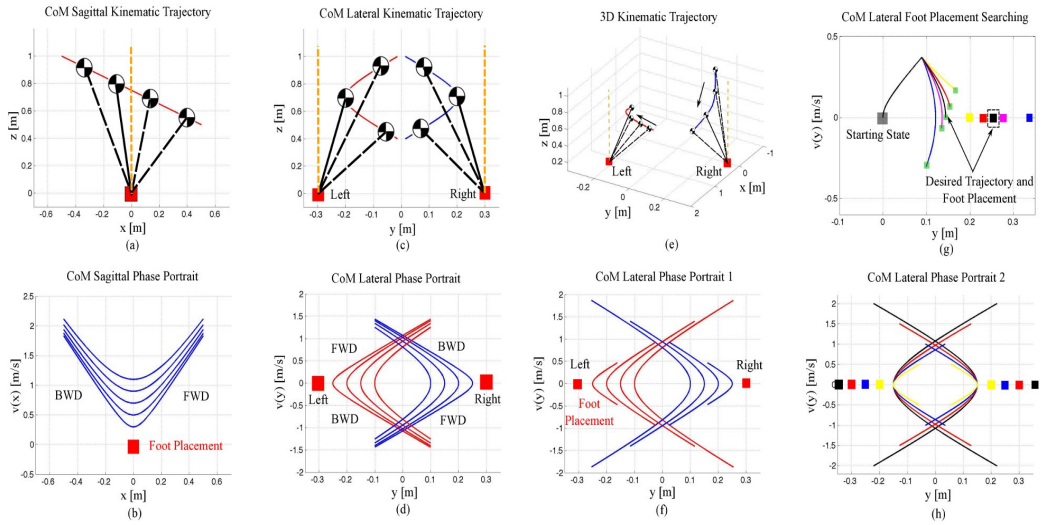


Figure 2.2: **Prediction of 3D single contact behaviors.** A prismatic inverted pendulum (i.e. one in which the height can change) is utilized to study the sagittal (a) and lateral (c) motion. In (a), the center of mass traverses the apex point while the center of mass in (c) bounces back before reaching the lateral foot position. The phase diagrams (b) and (d) correspond to the sagittal and lateral center of mass phase behaviors given desired feet contact locations (red boxes), a desired center of mass surface of motion, and initial position and velocity conditions. The combined 3D motion is integrated in (e). If we consider timing issues on the lateral plane as discussed in Chapter 4, we can derive two different trajectories shown in (f) and (h). (f) shows lateral CoM behaviors given a fixed lateral foot placement and varying starting conditions. (h) corresponds to CoM trajectories derived given varying lateral foot placements and a fixed starting conditions. In (g), we analyze lateral CoM trajectories with one varying step transition.

nonlinear in their most general case [58]. In the special case shown in this paper it turns out that the dynamics become linear. However, considering the nonlinear case, a closed form solution of the dynamic behavior cannot be obtained. To address this limitation, we develop numerical integration techniques to solve the model dynamics.

Suppose that we have a nonlinear differential equation for the scalar variable x, and with form

$$\ddot{x} = f(x, \dot{x}). \tag{2.15}$$

We assume that $\ddot{x}$ is approximately constant for small increments of time. We discretize the trajectory, $(x_{k+1}, \dot{x}_{k+1})$, and derive Taylor expansions for a small disturbance, ϵ, and for initial conditions $(x_k, \dot{x}_k, \ddot{x}_k)$ to get

$$\dot{x}_{k+1} \approx \dot{x}_k + \ddot{x}_k \epsilon, \tag{2.16}$$

$$x_{k+1} \approx x_k + \dot{x}_k \epsilon + 0.5\, \ddot{x}_k \epsilon^2. \tag{2.17}$$

From Eq. (2.16) we find the expression of the perturbation, $\epsilon \approx (\dot{x}_{k+1} - \dot{x}_k)/\ddot{x}_k$, and substituting in Eq. (2.17), with $\ddot{x}_k = f(x_k, \dot{x}_k)$, we get

$$x_{k+1} \approx \frac{\left(\dot{x}_{k+1}^2 - \dot{x}_k^2\right)}{2\, f\left(x_k, \dot{x}_k\right)} + x_k \tag{2.18}$$

which is the state-space approximate solution that we are looking for. The pipeline for finding state-space trajectories goes as follows: (1) choose a very small time perturbations ϵ, (2) given known velocities $\dot{x}_k$ and accelerations $\ddot{x}_k$, and using Eq. (2.16), we get the next velocity $\dot{x}_{k+1}$, (3) using Eq. (2.18) we get the next position x_{k+1}, (4) plot the points $(x_{k+1},\, \dot{x}_{k+1})$ in the Phase Plane.

We also notice, that we can iterate this recursion both forward and backward. If we iterate backward we then need to choose a negative perturbation ϵ.

In Figure 2.2, we depict various single contact scenarios of sagittal and lateral trajectories, their combined solution and the effect of changing lateral feet locations.

Chapter 3

Data Driven Modeling

In this chapter, implicit regression is proposed for our data driven modeling in Phase Space. First, basic theories and formulations about support vector regression (SVR) are introduced. Then explicit and implicit regression are compared with each other to leverage advantages of the latter one: infinity slope and loop behaviors. Finally, trajectory training is generated based on the inverted pendulum model and at last contact switching is solved in the Phase Space.

3.1 Background on Support Vector Regression

We introduce here the basic concepts and formulations involved in our implementation of support vector regression. One of the main advantages of SVR is that it can discriminate a model or trajectory without knowing its information a priori. Additionally, SVR can approximate high dimensional nonlinear systems and provide a robust fitting solution.

Given a training data set $\{(\mathcal{X}_1, y_1), (\mathcal{X}_2, y_2), ..., (\mathcal{X}_l, y_l)\} \subset \mathbb{R}^N \times \mathbb{R}$, a function $\Phi(\mathcal{X})$ can be obtained, with a predefined deviation for all training data. A non-zero ϵ deviation is required to avoid over-fitting. The hyperplane in

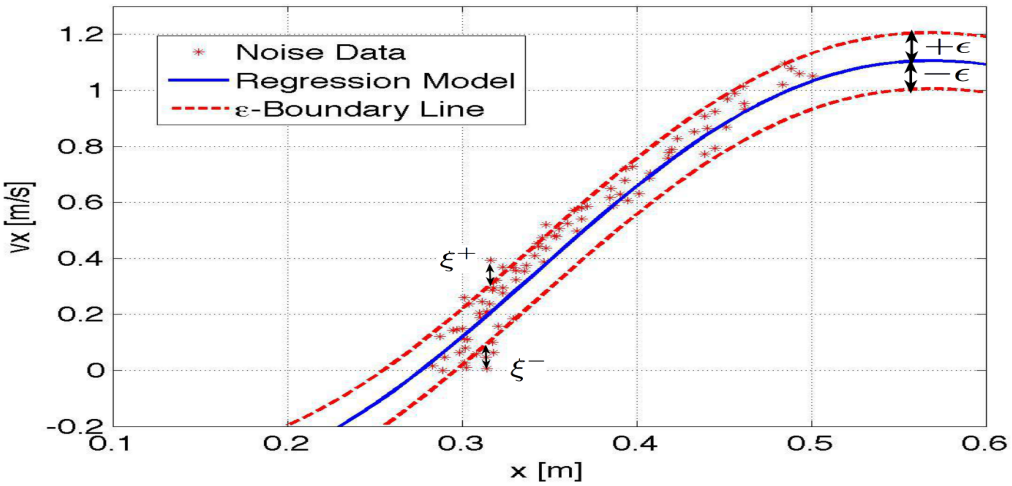


Figure 3.1: **Explicit support vector regression of inverted pendulum dynamics in phase space.** This figure shows SVR based on a Gaussian kernel (implemented using the software LibSVM [7]). Random noise is further added to the data (red star dots). The blue line is the nominal model trajectory after the training. The red dashed line is the ϵ-insensitive tube. Note that some data still exists outside the insensitive tube, which will cost some positive ξ errors. The SVR parameters used here are $C = 2, g = 10, \epsilon = 0.1, b = 0.2937$. The number of support vectors is 11 while the total number of data points is 88. The main defficiency of this explicit method is that the blue nominal trajectory does not reflect true pendulum dynamics, where the slope around $x = 0.3m$ has infinity slope. Infinity slope cannot be captured with explicit functions.

high dimensional feature space can be expressed as

$$\Phi(\mathcal{X}) = \langle \omega, \mathcal{X} \rangle + b \tag{3.1}$$

where $\mathcal{X}$ is the state, $\langle \cdot, \cdot \rangle$ represents the dot product in $\mathbb{R}^N$. $b \in \mathbb{R}$ is the intercept and $\omega \in \mathbb{R}^N$ denotes the normal vector to the hyperplane. As long as the errors are within an ϵ-insensitive tube, the fitting will not add a cost to the objective function. However, to deal with "bad" data appearing far away

from the insensitive region, slack variables ξ_i^+ and ξ_i^- are introduced to provide "Soft Margins". Based on the support vector machine tutorial describe in [66], we define the optimization problem

$$\begin{aligned} &\min_{\omega,b} \frac{1}{2}||\omega||^2 + C\sum_{i=1}^{l}(\xi_i^+ + \xi_i^-) \\ &\text{s.t. } -\epsilon - \xi_i^- \leq y_i - \langle\omega, \mathcal{X}_i\rangle - b \leq \epsilon + \xi_i^+, \\ &\qquad 0 \leq \xi_i^+, \xi_i^-. \end{aligned} \tag{3.2}$$

where ϵ is the model tolerance. The first inequality constraint guarantees that the function $\Phi(x)$ approximates the data pairs $(\mathcal{X}_i, y_i)$ with $\epsilon + \xi$ accuracy. The ϵ-insensitive loss function can be described by

$$|\xi|_\epsilon = \begin{cases} 0 & |\xi| \leq \epsilon \\ |\xi| - \epsilon & |\xi| > \epsilon \end{cases} \tag{3.3}$$

Note that, the coefficient C weights out the model complexity ($||\omega||$) versus the regression error ($\sum(\xi_i^+ + \xi_i^-)$). The structural risk minimization technique balances the model's complexity against its success in fitting the training data. By constructing Lagrange functions, the previous optimization problem can be expressed in the following manner (details are omitted)

$$\begin{aligned} \underset{\alpha,\alpha^*}{\text{maximize}} \quad & -\frac{1}{2}\sum_{i,j=1}^{l}(\alpha_i - \alpha_i^*)(\alpha_j - \alpha_j^*)\langle\mathcal{X}_i, \mathcal{X}_j\rangle \\ & -\epsilon\sum_{i=1}^{l}(\alpha_i + \alpha_i^*) + \sum_{i=1}^{l} y_i(\alpha_i - \alpha_i^*) \\ \text{subject to} \quad & \sum_{i=1}^{l}(\alpha_i - \alpha_i^*) = 0, \alpha_i, \alpha_i^* \in [0, C]. \end{aligned} \tag{3.4}$$

where $\mathcal{X}_i$ is the i^{th} support vector and l is the total number of support vectors. α and α^* are Lagrange multipliers of the first inequality constraint in

Equation (3.2) and the normal vector $\omega = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)x_i$. Then solving for Equation (3.1), the output response becomes

$$\Phi(x) = \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)k(\mathcal{X}_i, \mathcal{X}) + b \tag{3.5}$$

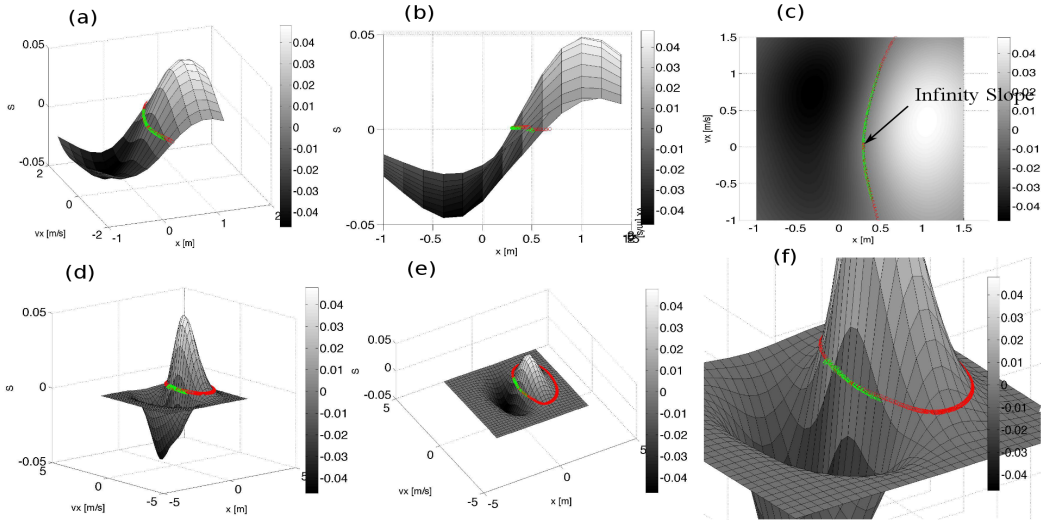


Figure 3.2: **2D implicit regression with gaussian kernel.** The SVR parameters are selected as $C = 2, g = 1, \epsilon = 0.02, b = 0.0073$. The number of support vectors is 87. Subfigures (a) - (c) show the fitting surface from different viewpoints. Green is the data points obtained through simulation of a nonlinear prismatic pendulum while red is the surface fitting. Notice that in (c) we demonstrate that the fitting is accurate even when the slope is infinity on the $(x, \dot{x})$ plane. Subfigures (d) - (f) show the same surface with a longer range of values.

There are several choices for the kernel function. For our study, we choose a Gaussian kernel

$$k(x, x') = exp(-\gamma||\mathcal{X} - \mathcal{X}'||^2) \tag{3.6}$$

where γ defines the Gaussian kernel width. Notice that sigmoid and polynomial kernels are also popular for curve fitting.

3.2 SVR-Based Motion Planning

We approximate single contact dynamics using a nonlinear prismatic pendulum and then use numerical simulation to obtain the training data. We will show the main advantage of using implicit versus explicit functions for regression. First, let us consider an explicit regression case with two dimensional states.

3.2.1 Explicit Regression

The trained model has the following expression

$$\dot{x} = \Phi(x) = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) k(x_i, x) + b, \tag{3.7}$$

where x and $\dot{x}$ are the position and velocity states respectively. In particular, since we are dealing with Phase Space trajectories, the velocity is chosen as the function output of the regression process.

In Fig. 3.1, we show an example of fitting data points from the pendulum simulation using the above explicit function. Because the number of support vectors is monotonically decreasing with ϵ, the larger it is, the smaller the number of support vectors are needed, thus reducing the computational effort. On the other hand, too large ϵ will lead to under-fitting. So choosing the right ϵ value is key to the solution. However, there exists major drawbacks to using explicit functions. First, they cannot represent loop dependencies, such as a circular motion. Loop behaviors in Phase Space are characteristics of locomotion patterns. Second, the general explicit solution given below (for

a Gaussian kernel)

$$\frac{d\dot{x}}{dx} = -2\gamma \sum_{i=1}^{l} (\alpha_i - \alpha_i^*)(x - x_i) exp(-\gamma(x - x_i)^2) \tag{3.8}$$

shows that the slope can never be infinity since it is the sum of finite positive numbers. However, one of the properties of pendulum dynamics is having infinity slope when changing the velocity sign. Moreover, explicitly functions cannot represent cyclic loops that are typical of Phase Space trajectories. Given those limitations, we propose to use implicit regression instead.

3.2.2 Implicit Regression

The trained model has the following expression

$$\Phi(\mathcal{X}) = \sum_{i=1}^{l} (\alpha_i - \alpha_i^*) k(\mathcal{X}_i, \mathcal{X}) + b \tag{3.9}$$

where the implicit state is $\mathcal{X} = (x, \dot{x})$ and $\Phi(\mathcal{X})$ is a continuous differentiable function defined as the surface s. Here, it can be seen that compared with explicit regression, the velocity is included into the state variables and coupled with the position state x. The partial derivatives are now

$$\frac{\partial \Phi(\mathcal{X})}{\partial x} = -2\gamma \sum_{i=1}^{l} (\alpha_i - \alpha_i^*)(x - x_i) A_{exp} \tag{3.10}$$

$$\frac{\partial \Phi(\mathcal{X})}{\partial \dot{x}} = -2\gamma \sum_{i=1}^{l} (\alpha_i - \alpha_i^*)(\dot{x} - \dot{x}_i) A_{exp} \tag{3.11}$$

where $A_{exp} \triangleq exp(-\gamma((x - x_i)^2 + (\dot{x} - \dot{x}_i)^2))$. The output response can now be considered a trajectory surface and therefore displayed as a third dimension

besides the position and velocity states. In Fig. 3.2 we use once more prismatic pendulum dynamics to generate data points and use Equation (3.9) to fit the surface. We then display the three dimensions from various perspectives. As we can see, we are now able to fit the data even if the slope is infinity at some point. Moreover, the same implicit method could be used to fit loop trajectories typical of locomotion in the Phase Plane.

3.2.3 Data Generation

The data points shown in Figs. 3.1 and 3.2 can be obtained in several ways. In an ideal scenario, the real robot could be prompted to move along predefined geometric paths and data points of the center of mass position and velocity could be recorded for every path. However, for this case, we will use a simplified model of the dynamics and training data is generated from a simulation. In Chapter 2, we have approximated the single contact dynamics using a prismatic inverted pendulum with variable height [78] to yield center of mass dynamics of form

$$\ddot{x} = \frac{(x - p_x)(z + g)}{z - p_z} \tag{3.12}$$

where x and z are the sagittal and vertical coordinates of the center of mass and p_x and p_z are the sagittal and vertical coordinates of the support leg. For geometric primitives that are linear, e.g. $z = ax + b$, the above equation can be further evolved into

$$\ddot{x} = \frac{(x - p_x)g}{ap_x + b - p_z} \tag{3.13}$$

which defining the constant $\alpha = \sqrt{\frac{g}{ap_x+b-p_z}}$ can be written as a linear dynamical system of form

$$\ddot{x} = \alpha^2(x - p_x) \tag{3.14}$$

This model is convenient because it can be solved analytically. However, in the past we have tackled this problem via solving the dynamics via numerical simulation [62]. For this study, we use the above linear form to obtain data points. If one was to solve the above equation through some standard linear method, e.g. Laplace, we would get the close-form solution

$$x(t) = (x_0 - p_x)\cosh(\alpha t) + \frac{1}{\alpha}\dot{x}_0\sinh(\alpha t) + p_x \tag{3.15}$$

$$\dot{x}(t) = \alpha(x_0 - p_x)\sinh(\alpha t) + \dot{x}_0\cosh(\alpha t) \tag{3.16}$$

Using the trigonometric equality $\cosh^2(x) - \sinh^2(x) = 1$, we can remove time from the above equations and obtain the implicit surface

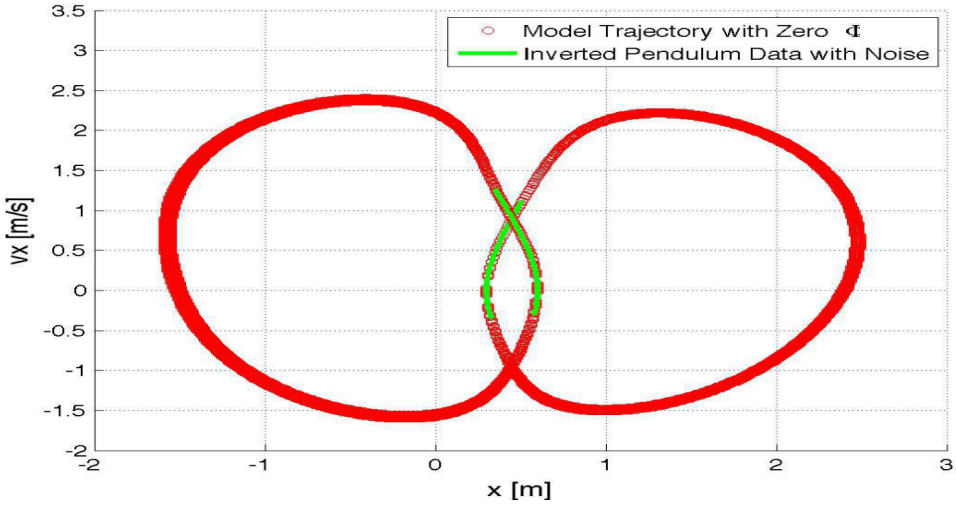
$$\begin{aligned}\Phi(x, \dot{x}) = (x_0 - p_x)^2(2\dot{x}_0^2 - \dot{x}^2 + \alpha^2(x - x_0)(x + x_0 - 2p_x)) \\ - \dot{x}_0^2(x - p_x)^2 + \dot{x}_0^2(\dot{x}^2 - \dot{x}_0^2)/\alpha^2\end{aligned} \tag{3.17}$$

Since this two dimensional hyper-surface is a differentiable manifold in 3D space, the surface normal can be represented by the gradient

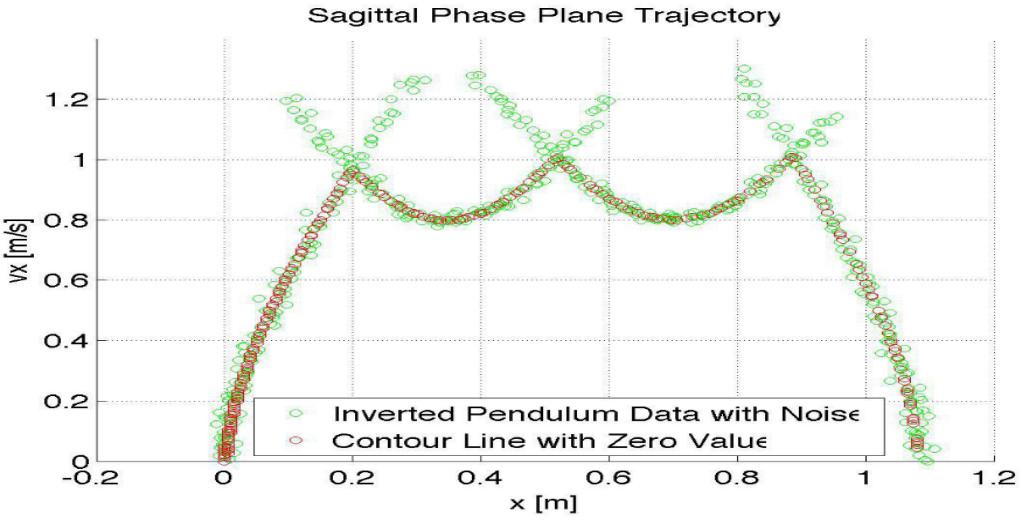
$$\mathcal{N} \triangleq \left[\frac{\partial\Phi}{\partial x}, \frac{\partial\Phi}{\partial\dot{x}}, 1\right]^T \tag{3.18}$$

Using Equations (3.10) and (3.11) and projecting the normal vector in the horizontal plane yields

$$\frac{d\dot{x}}{dx} = \frac{\partial\Phi/\partial x}{\partial\Phi/\partial\dot{x}} = \frac{\sum_{i=1}^{l}(\alpha_i - \alpha_i^*)(x - x_i)A_{exp}}{\sum_{i=1}^{l}(\alpha_i - \alpha_i^*)(\dot{x} - \dot{x}_i)A_{exp}} \tag{3.19}$$



(a) Single Step Planner



(b) Multi Step Planner

Figure 3.3: **Single and multi step planners.** (a) Illustrates the intersection point of two adjacent regression trajectories. The green and red circles correspond to the implicit functions that represent the black and blue data sets, respectively. The intersection between the black and blue curves represents the contact event that will stabilize the hybrid dynamics (i.e. the step). (b) Illustrates multi step planning in the sagittal plane. The red data is the contour line when the model surface Φ is zero while the green data is used for regression. More details about this planner will be studied in Chapter 4.

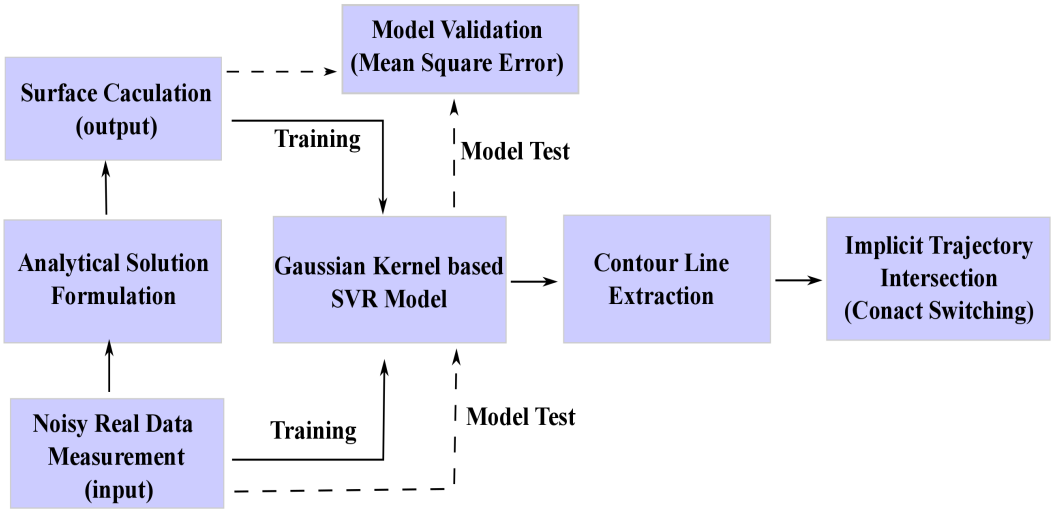


Figure 3.4: **Support vector regression pipeline diagram.** Here, the block "contour line extraction" is used for extracting the model dynamics which corresponds to the surface being equal to zero. The block "implicit trajectory intersection" is our contact switching policy, which means extracting the intersection point of two adjacent implicit regression trajectories in Phase Space.

when $\dot{x} \to 0$ it becomes,

$$\frac{d\dot{x}}{dx} = \frac{\sum_{i=1}^{l}(\alpha_i - \alpha_i^*)(x - x_i)exp(-\gamma((x - x_i)^2 + \dot{x}_i^2))}{-\sum_{i=1}^{l}(\alpha_i - \alpha_i^*)\dot{x}_i exp(-\gamma((x - x_i)^2 + \dot{x}_i^2))} \tag{3.20}$$

It turns out that the denominator of the above equation becomes very small when changing velocity signs, thus illustrating that pendulum dynamics often yields infinite slopes, as shown in Fig. 3.2 (c).

3.2.4 Contact Switching Policy

Once the implicit surfaces have been generated based on the data points, we extract contour lines defined as

$$\mathcal{H}_c(\Phi) \triangleq \big\{(x, \dot{x})|\Phi(x, \dot{x}) = 0)\big\} \tag{3.21}$$

The solution for extracting the intersection point of two adjacent implicit regression trajectories can be easily solved using a nonlinear equation solver (e.g. fsolve() in Matlab). If we name the adjacent surfaces $\Phi_1(\mathcal{X})$ and $\Phi_2(\mathcal{X})$, all we need to do is solve the nonlinear equation

$$F(\mathcal{X}) = 0 \tag{3.22}$$

with

$$F(\mathcal{X}) \triangleq \Phi_1(\mathcal{X}) - \Phi_2(\mathcal{X}), \Phi_1(\mathcal{X}) = 0 \tag{3.23}$$

The solution of the above differential will be the desired contact event that will stabilize the locomotion behavior. A one step example and a multi step example are shown in Figs. 3.3 (a) and (b). The pipeline for finding contact points is summarized in the block diagram of Fig. 3.4.

Chapter 4

Phase Space Planning

In this chapter, a foot placement searching strategy is formulated in Phase Space based on the inverted pendulum in Chapter 2. Newton-Raphson method is used to search lateral foot placement while satisfying sagittal timing constraint. In addition, multi-contact transition phase are considered at the same time. Then we show the capability of our walking planner to maneuver the rough terrain agilely. Last but not least, a new searching strategy is presented based on a modified 3D inverted pendulum model. In this scenario, both sagittal and lateral foot placements are searched simultaneously in a nested manner. More details about guaranteed walking symmetries and zero lateral velocity at foot apex are shown in the searching algorithm.

4.1 3D Foot Placement Planner

In our previous studies [58, 61], our method successfully predicted the phase curves of center of mass sagittal behavior and was used to find the solutions of step transitions as the intersections between adjacent phase curves. See Figure 4.1 for a depiction of sagittal feet placements.

In this chapter, our main focus is on the extension of our solver to the

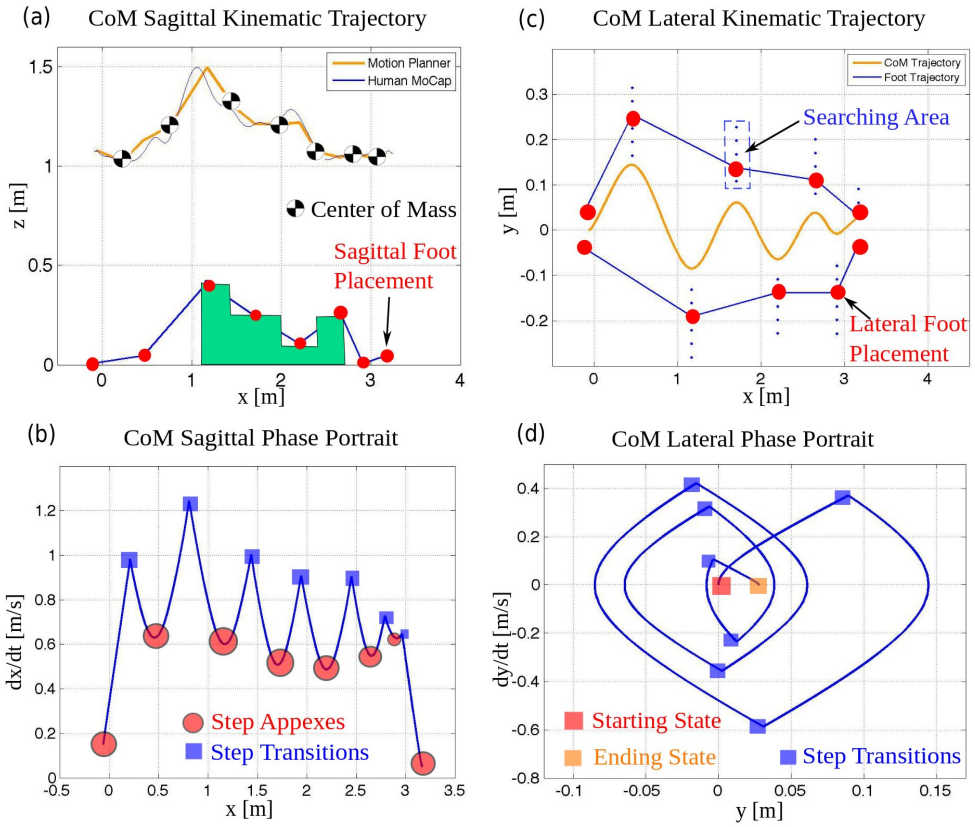


Figure 4.1: **3D automatic motion planner.** (a) corresponds to the user-defined geometric trajectory of the center of mass on the sagittal plane and desired sagittal feet locations, while (b) corresponds to the phase plane output of the proposed motion planner. Given step apex conditions (i.e. positions and velocities when crossing the apex), single contact dynamics generate the valley profiles shown in (b). Our planning strategy is to find intersection between adjacent contact behaviors which ensure continuity on positions and velocities. To obtain the intersections, we fit polynomials to the phase behaviors and find the roots of the polynomial resulting from subtracting adjacent curves. (d) depicts a similar strategy in the lateral plane. However, since feet transitions have already been determined in (b), what is left is to determine feet lateral positions as shown in (c). This is done so the lateral center of mass behavior shown in (d) follows a semi-periodic trajectory that is bounded within reasonable values.

lateral motion plane. As such, it will allow us to create 3D gait plans. This problem is difficult because once we have determined feet sagittal transitions, we are committed to a foot step timing. Therefore we develop a new search strategy that enables to find feet placements in the lateral plane that comply with the timing constraints.

4.1.1 Lateral Single Contact Behavior using Sagittal Timing

Similar to the sagittal case, numerical integration is used to determine phase curves. However, in the lateral case we do not know the reset condition at every step (i.e. lateral velocities at known points) since walking velocities are only specified sagittally. Instead, the main objective of lateral behavior is to produce bounded semi-periodic trajectories. Because we do not know phase plane points in the lateral plane, we rely on forward propagation of Equation (2.14) that complies with the timing constraints. This technique is shown in Figure 2.2 (g), where multiple curves are shown that complete the timing cycle of the sagittal planner but change depending on the lateral placement locations. Implementing this idea for multiple steps leads to semi-periodic gait sequences such as the one shown in Figure 4.1. Specifically, the blue squares correspond to the points where two curves from neighboring steps have the same position and velocity and therefore correspond to feet lateral transitions. Also, on the top right plot of the same figure we illustrate the need to search over multiple lateral locations to ensure that the trajectories are bounded.

The lateral phase portrait behaves like a semi-periodic cycle, but several differences exist resulting from the sharp contact transitions in the uneven terrain. At every step transition, a "sharp corner" appears due to the drastic change of acceleration. We will show next the need to smoother these corners by introducing multi-contact phases.

4.1.2 Searching Strategy for Lateral Foot Placement

As previously shown, the 3D dynamics are broken into sagittal and lateral behavior, which are each separately solved for. However, to unify the 3D foot placement planner, the time spent during each step should be the same on both simulations. As in our previous studies [58, 61], we first solve for the sagittal feet transitions using forward and backward numerical integration and given the apex conditions (see Figure 4.1). Next, it is straightforward to solve for the lateral behavior using forward numerical integration and then switching contact models at exactly the same time as the feet transitions derived from the sagittal planner. The problem with this technique, is that lateral feet locations will dramatically influence the phase trajectory. If lateral feet placements are not adequately picked, the lateral behavior will not produce a bounded trajectory cycle and therefore the steps will drift away, ultimately becoming unstable.

In Equation (2.14) we show the direct dependencies of lateral behavior with lateral feet placements, and the plot of Figure 2.2 (g) shows the impact of using different placements. The question is, which foot placement is the

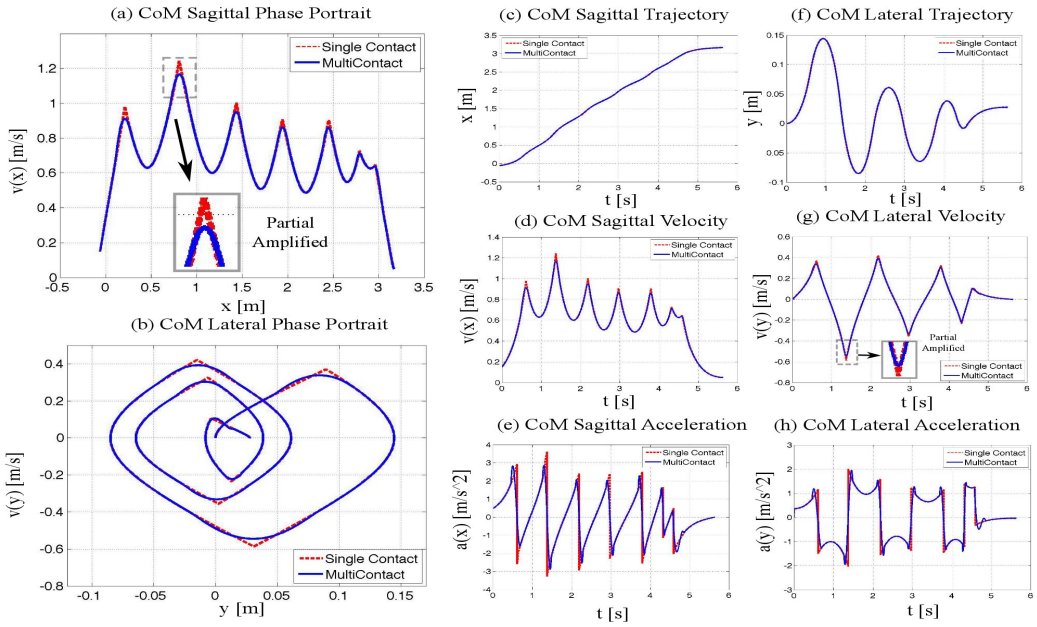


Figure 4.2: **Integration of multi-contact phases.** The plots (a) and (b) are similar to their counterparts of Figure 4.1 but with an addition of a multi-contact phase. A user decides the duration of the multi-contact phase with respect to the overall step and then chooses the velocity and acceleration profile during multi-contact. By using 5th order polynomials and guaranteeing continuity with the existing curves, we get the polynomial parameters and fit the curve. To determine the feasibility of the curves we extract internal forces using the multi-contact-grasp matrix presented in [61] and then determine if they are feasible given surface friction constraints. Plots (c) through (h) depict the time profiles for the sagittal and lateral trajectories.

Algorithm 1 Newton-Raphson Search for Lateral Foot Placement

Assign iteration step $k = 1$
Choose the initial value $(FP_y)_1$
while $k < IterationNum_{max}$ and $(v_{com_y}(\text{end}))_k > VY_{tolerance}$ **do**
  Implement numerical integration with $(FP_y)_k$ for one step and obtain $(v_{com_y}(\text{end}))_k$
  Derive $(FP_y)_{k+1}$ by Newton-Raphson Formula in (4.1)
  Implement numerical integration of $(FP_y)_{k+1}$ for one step and obtain $(v_{com_y}(\text{end}))_{k+1}$
  $(a_{com_y}(\text{end}))_{k+1} = \frac{(v_{com_y}(\text{end}))_{k+1} - (v_{com_y}(\text{end}))_k}{(FP_y)_{k+1} - (FP_y)_k}$
  $k = k + 1$
end while

best option? An ad-hoc choice is to choose the one that produces zero lateral velocity when the center of mass crosses the sagittal apex of the foot. Based on this criterion, we implement a foot placement search strategy based on the Newton-Raphson bisection method, i.e.

$$(FP_y)_{k+1} = (FP_y)_k - \frac{(v_{com_y}(\text{end}))_k}{(a_{com_y}(\text{end}))_k} \tag{4.1}$$

where $(FP_y)_{k+1}$ are the candidate lateral feet placement locations for the k^{th} incremental search, $(v_{com_y}(\text{end}))_k$ and $(a_{com_y}(\text{end}))_k$ represent the final velocity and acceleration achieved in the previous search. For simplicity, $(a_{com_y}(\text{end}))_k$ is obtained via numerical differentiation. The objective of the algorithm is to iterate over $(FP_y)_{k+1}$ until $(v_{com_y}(\text{end}))_k$ is sufficiently close to zero. Overall, the search algorithm goes as follows

The result of using the above algorithm can be seen in the lower right plot of Figure 4.1.

4.1.3 Multi-Contact Transitions

Without multi-contact [29, 59, 61], contact transitions cause discontinuities in the sagittal and lateral behaviors. Moreover, such transitions are unrealistic as robots cannot switch feet instantaneously. It is also not desirable to switch feet too quickly to prevent reaching high velocity peaks (see Figure 4.2 (a)). Our objective here is to incorporate multi-contact transitions into our gait planner to make it look more natural and to reduce velocity peaks. For this purpose, we augment our planner with a multi-contact phase.

To incorporate a multi-contact phase, we cut out a portion of the phase curves and fit a polynomial with the desired smooth behavior. In this fitting, desired boundary values of position, velocity and acceleration are endowed by the gait designer. More importantly, it is needed to also take into account time constraints in such a way that the sagittal and lateral behaviors are exactly synchronized. Boundary and timing conditions allow us to calculate the coefficients of the polynomials.

For instance, if the polynomial of the multi-contact phase is defined by the formula

$$x(t) = \sum_{i=0}^{5} a_i \, t^i \tag{4.2}$$

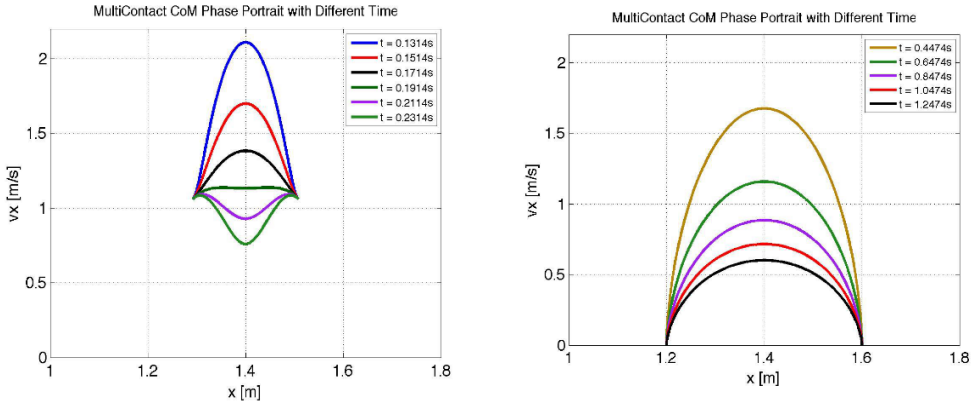


Figure 4.3: **Multi-contact polynomial fitting.** The propagation time reduces as CoM trajectories move from top to bottom. In subfigure on right, it shows similar time properties with zero initial and final velocity.

.

then we can calculate its coefficients as

$$a_0 = x_{mi}, \quad a_1 = v_{mi}, \quad a_2 = \frac{a_{mi}}{2}, \tag{4.3}$$

$$a_3 = \frac{1}{2t_f^3}\left(20x_{mf} - 20x_{mi} - (8v_{mf} + 12v_{mi})t_f - (3a_{mi} - a_{mf})t_f^2\right), \tag{4.4}$$

$$a_4 = \frac{1}{2t_f^4}\left(30x_{mi} - 30x_{mf} + (14v_{mf} + 16v_{mi})t_f + (3a_{mi} - 2a_{mf})t_f^2\right), \tag{4.5}$$

$$a_5 = \frac{1}{2t_f^5}\left(12x_{mf} - 12x_{mi} - (6v_{mf} + 6v_{mi})t_f - (a_{mi} - a_{mf})t_f^2\right). \tag{4.6}$$

where $[.]_{mi}$ and $[.]_{mf}$ are initial and final position, velocity and acceleration conditions, and t_i and t_f are initial and final times.

In our case, we design a multi-contact phase that takes place during 25% of the time of any given step. We show the result in Figure 4.2. The dotted rectangle in plot (a) of the previous Figure depicts the time window for

multi-contact. This percentage is adjustable to other values based on the desired walking profile. Not done here, the multi-contact profile entails internal forces that can be derived using the techniques that we proposed in [61]. The resulting internal forces would then need to be validated against the friction characteristics of the terrain. Additionally, shapes of polynomial fitting trajectories are varied when the time period is changed. We generate a family of phase plane trajectory candidates during the multi-contact phase and analyze its characteristics in Fig. 4.3.

4.1.4 Time Trajectory Generation

Our planner relies on input data sets that include: (1) the motion surface of the center of mass, (2) sagittal feet placement locations, and (3) desired sagittal velocities at the apex points. Given these data, the planner determines: (1) sagittal and lateral center of mass phase curves, (2) lateral feet locations, and (3) transition points of the feet in the phase plane. These data needs to be converted to time trajectories. Since the center of mass curves are continuous, it is straightforward to convert them into a time trajectory. On the other hand, because the feet transitions are discrete, we interpolate smooth leg swinging trajectories to land the feet at the desired time stamps. Finally, we use inverse kinematics to fit the robot's multi-joint structure to the desired trajectories. We do this process for our case study and display it in the animation of Figure 4.4.

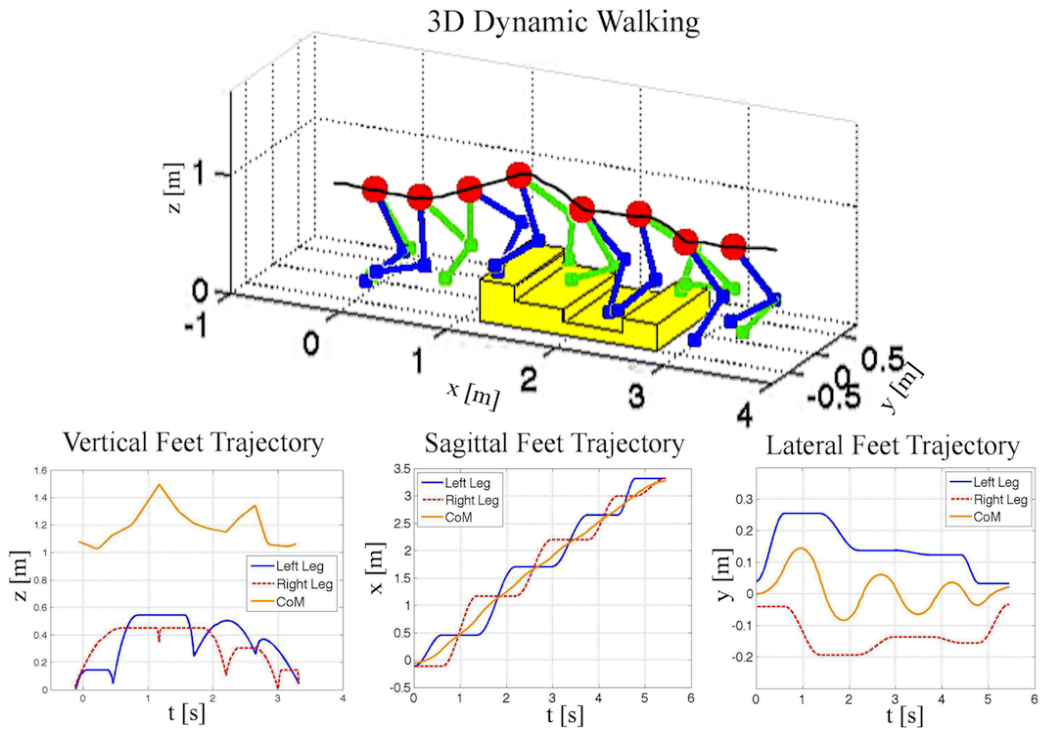


Figure 4.4: **3D dynamic walking animation.** (a) shows an animation of the robot executing the planned trajectories. To display the results on a legged model, we fit continuous feet trajectories that converge to the desired contact conditions and we run an inverse kinematic process to obtain the resulting joint angles. Since the feet are Cartesian points we need to plan trajectories in the vertical (b), sagittal (c), and lateral axes (d).

4.1.5 Dynamic Maneuvering on Different Terrains

Without loss of generality, our algorithm is implemented for three more challenging terrains shown in Figure 4.5: inclined terrain, concave terrain and convex terrain. The walking on inclined terrain is illustrated in sequential snapshots from three different viewpoints. In this case, the height discrepancy of two consecutive stairs is specified to be a random value with a maximum

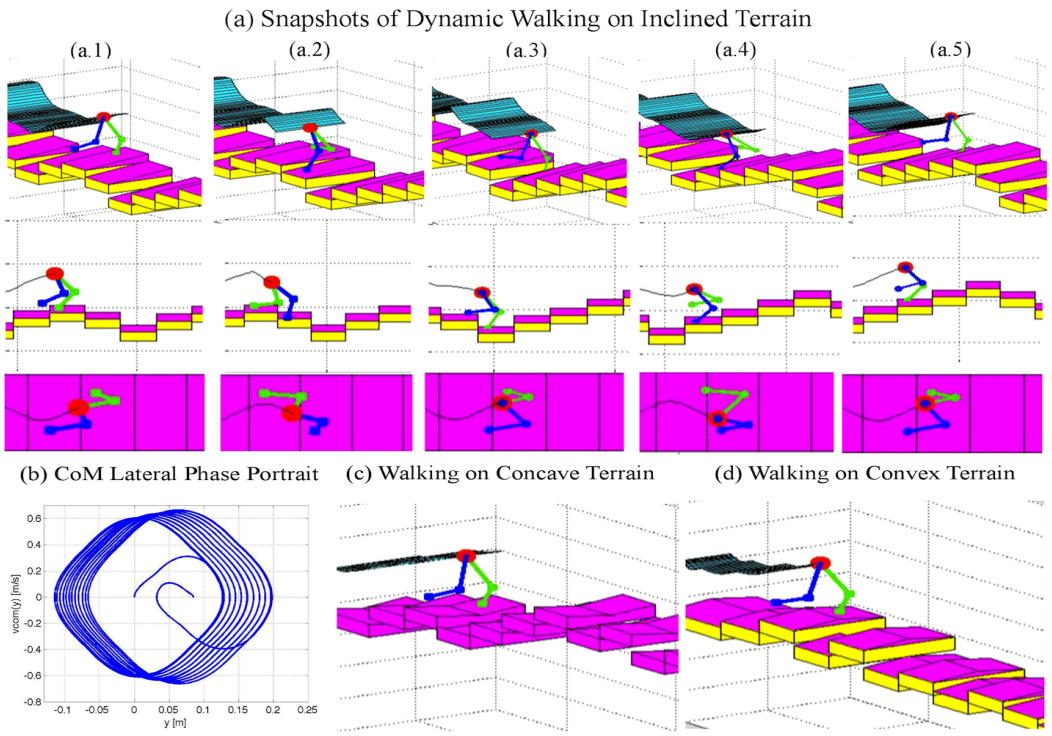


Figure 4.5: **Traversing of different terrain profiles.** (a) shows the snapshots of walking on a terrain with rough inclined surfaces. The side and top visualization illustrates the agile walking capabilities. The lateral CoM phase portrait in (b) shows a 25 steps walking sequence. In (c) and (d) we test the applicability of our gait generator to various terrains.

of 0.2 meters. A 10 degree tilt angle is assigned to the slope of the surface. Finally, the planner generates 25 steps as shown in Figure 4.5 (b). Our algorithm is also tested on two different inclined terrains shown in Figure 4.5 (c) and (d). The average walking speed is 0.8m/s. These visualizations indicate the applicability of dynamic maneuvering on challenging terrains.

4.2 Modified 3D Inverted Pendulum Model

The result above assumes the CoM is constrained to a plane described by a piece-wise linear function $z = a_i x + b_i, i = 1, ..., N$, which is a 2D plane in three dimensional space. As to the study in this section, this CoM plane assumption is extended to a 3D plane in three dimensional space.

4.2.1 3D CoM Plane Tracking

First, let us describe this new 3D plane mathematically

$$\lambda_1 x + \lambda_2 y + \lambda_3 z = \lambda_4 \tag{4.7}$$

where (x, y, z) is the CoM position state. Then

$$z = \frac{-\lambda_1}{\lambda_3} x + \frac{-\lambda_2}{\lambda_3} y + \frac{\lambda_4}{\lambda_3} = a_1 x + a_2 y + b \tag{4.8}$$

Differentiate it twice, we have

$$\ddot{z} = a_1 \ddot{x} + a_2 \ddot{y} \tag{4.9}$$

Based on the moment and force equations we have, one has

$$\ddot{x} = \frac{(x - p_x)(\ddot{z} + g)}{z - p_z}, \tag{4.10}$$

$$\ddot{y} = \frac{(y - p_y)(\ddot{z} + g)}{z - p_z}, \tag{4.11}$$

$$\ddot{y} = \frac{(y - p_y)\ddot{x}}{x - p_x} \tag{4.12}$$

Substitute Equation (4.9) into Equations (4.10) and (4.11), then we have

$$\ddot{x} = \frac{(x - p_x)(a_1\ddot{x} + a_2\ddot{y} + g)}{a_1 x + a_2 y + b - p_z}, \tag{4.13}$$

$$\ddot{y} = \frac{(y - p_y)(a_1\ddot{x} + a_2\ddot{y} + g)}{a_1 x + a_2 y + b - p_z} \tag{4.14}$$

By Equations (4.12) and (4.13), one has

$$\ddot{x} = \frac{(x - p_x)g}{a_1 p_x + a_2 p_y + b - p_z} = f_x(x, p_x, p_y, p_z) \tag{4.15}$$

By Equations (4.12) and (4.14), it shows

$$\ddot{y} = \frac{(y - p_y)g}{a_1 p_x + a_2 p_y + b - p_z} = f_y(y, p_x, p_y, p_z) \tag{4.16}$$

The benefit of this modified model lies in that the CoM is parameterized as a 3D plane instead of 2D plane. This provides more freedom for CoM motion pattern.

4.2.2 Nested-loop Searching Strategy

For this modified model, our foot planner strategy is to search both sagittal and lateral foot placement at the same time. In our algorithm, there are two nested searching loops, where sagittal foot placement is searched in an outer loop while lateral foot placement is searched in an inner loop. The outer loop (sagittal foot placement) guarantees walking symmetry, (i.e., switching foot

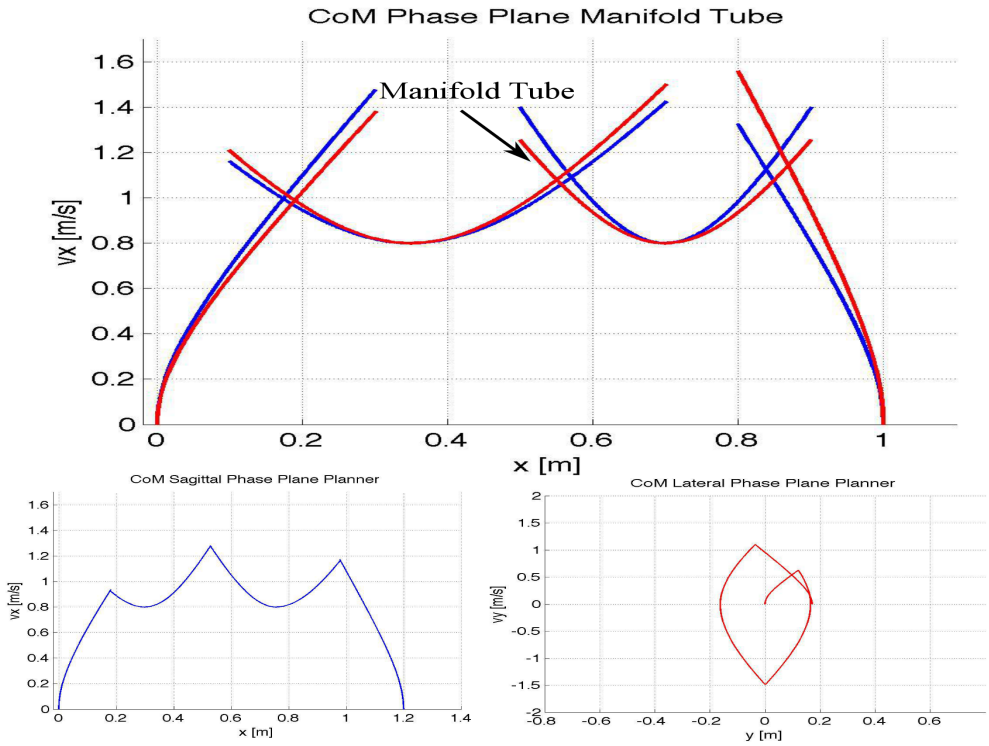


Figure 4.6: **New phase plane trajectories for three steps walking.** In subfigure (a), dynamics in sagittal phase plane vary for different lateral foot placements. Accordingly, a robust phase plane manifold tube appears. In subfigures (b) and (c), we use this new two dimensional foot placement searching strategy in Algorithm 2.

when the CoM approximately crosses central walking line) while the inner loop (lateral foot placement) guarantees zero-apex lateral velocity. This searching algorithm is presented in Algorithm 2.

As shown in Equations (4.15) and (4.16), the sagittal dynamics depend on lateral foot position p_y while the lateral dynamics also depend on sagittal foot position p_x. That is, if lateral (or sagittal) foot positions are assigned with different values, the sagittal (or lateral) phase trajectory will vary. Cor-

Algorithm 2 Foot Placement Searching Policy

Inverted Pendulum Following 3D Surface
$\ddot{x} = (x - p_x)g/(a_1 p_x + a_2 p_y + b - p_z) = f_x(x, p_x, p_y, p_z)$
$\ddot{y} = (y - p_y)g/(a_1 p_x + a_2 p_y + b - p_z) = f_y(y, p_x, p_y, p_z)$
Searching Region Initialization
Assign values of $foot_{x_{min}}, foot_{x_{max}}, foot_{y_{min}}, foot_{y_{max}}$
Foot Placement Initialization
Assign values of p_x, p_y
3D Plane Parameterization and Apex CoM Assignment
$a_1, a_2, (x, y, z)_{apex} \Rightarrow b$ is computed from Equation (4.8)
while $abs(y_{switch}) < Y_{tolerance}$ **do**
  (Outer loop guarantees the walking symmetry)
  Update sagittal foot placement
  while $abs(v_{y_{apex}}) < VY_{tolerance}$ **do**
    (Inner loop guarantees zero lateral velocity at foot apex)
    Update lateral foot placement
    Sagittal phase plane: Find intersection point $(x, \dot{x})_{intersect}$
    Lateral phase plane: Forward numerical integration to obtain $v_{y_{apex}}$.
    Bisection method to update the **lateral** foot placement
  end while
  Bisection method to update the **sagittal** foot placement
end while

respondingly, a manifold phase plane tube will show up in Fig. 4.6. In fact, the real CoM motion is constrained by robot mechanical configuration, motor torque limitation and friction cone, etc. These factors will formulate a bounded feasible region in the phase plane. In future work, we will study the phase plane manifolds restricted by these physical constraints.

Compared with our previous model, the model changes can be seen as follows. First, the CoM follows a 3D plane instead of a 2D plane. Second, for this modified model, there is no master-slave relationship between sagittal-lateral

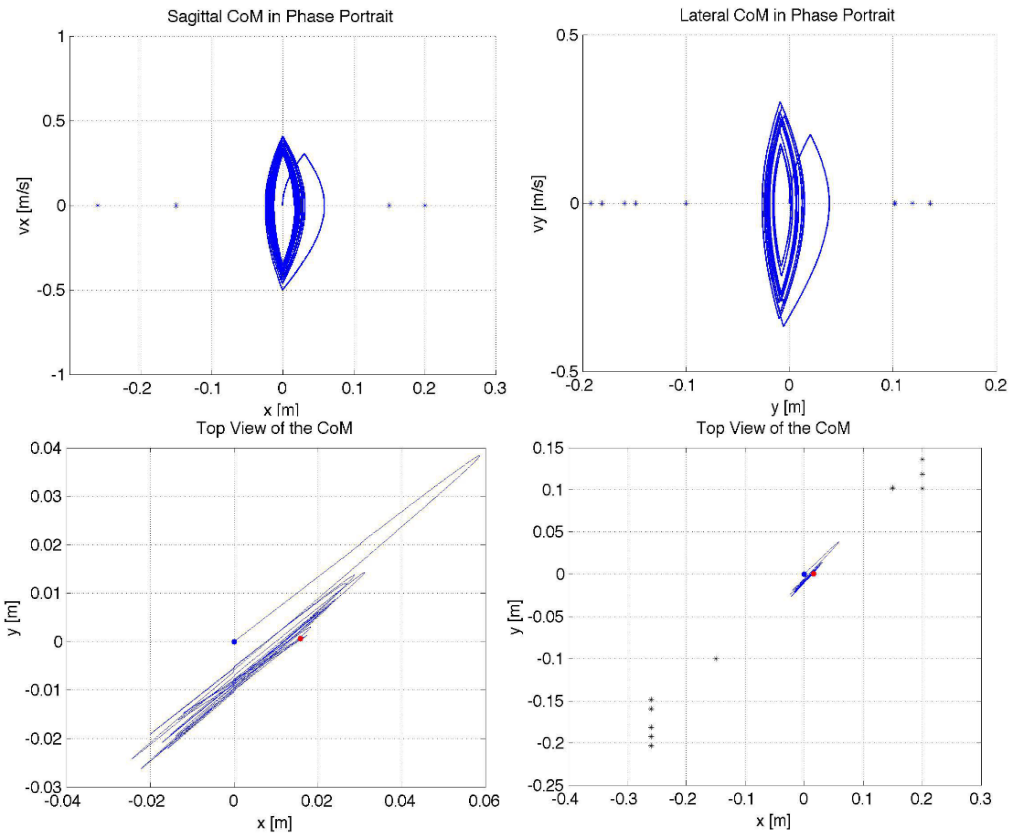


Figure 4.7: **One step balancing.** The figures here show the biped balancing using the modified 3D model searching strategy. It balances back and forth 20 times. The beginning step is quite different from the remaining ones.

dynamics, like the previous model in this Chapter. Both sagittal and lateral foot placement are searched simultaneously. This two-dimensional search is a more realistic scenario, because of its consistency with the fact that inverted pendulum dynamics are identical in either sagittal and lateral direction.

4.2.3 One Step Balancing

In this section, we implement our new algorithm to a one-step back-and-forth balancing case. Since our bipedal robot has point feet, this one-step balancing is more realistic than the one-step stop scenario proposed by Capture point [33]. A simulation shows 20-steps balancing in Fig. 4.7.

Note that, our data driven modelings haven't implemented this new searching algorithm till now. In the next phase, we will transfer to this modified model and derive the new 3D locomotion strategy.

Chapter 5

Robust Control

This chapter is about design of a robust control approach for locomotion based on our previous dynamic planner. In Chapter 3, we developed tools to accurately design a contact switching policy for stable locomotion. And we did so using sagittal pendulum dynamics because they are dominant in the production of net movement. However, in biped locomotion, center of mass dynamics are difficult to control due to the small size of the feet. They can be considered as passive dynamics. The purpose of locomotion is then to control precisely the swinging leg, so it switches dynamics when contact occurs. Those needs where studied at the conceptual level in previous chapters.

5.1 Sliding Mode Control

Let us assume for now, that a hybrid phase plan has been designed for desired foot positions and according to the methods described in the chapter 3. Given the planned contact transitions, we subsequently extract contact times from the Phase Space plan. Moreover, once the contact times are known, time-based foot trajectories can be easily designed. An example of a time based trajectory to swing toward a step is shown in Fig. 5.1. A valid con-

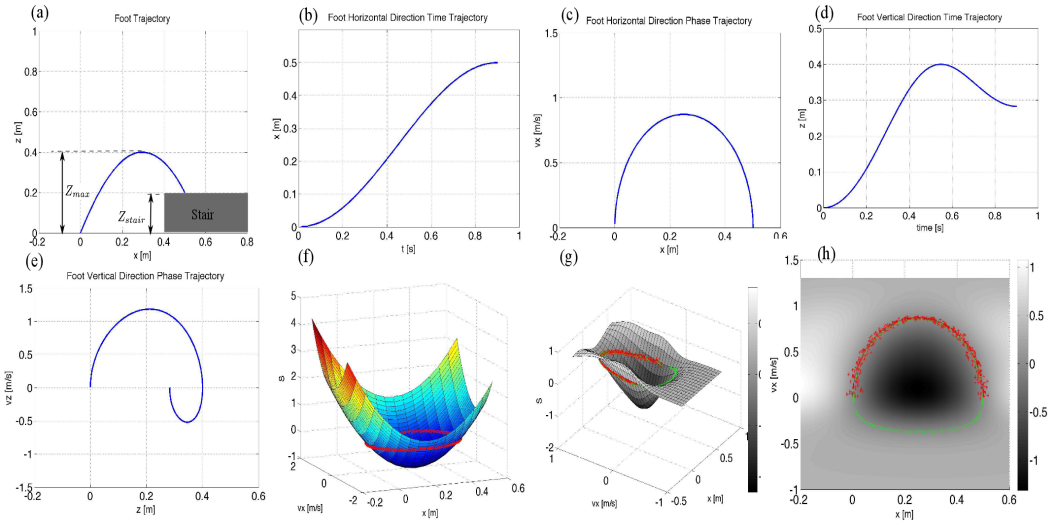


Figure 5.1: **Two dimensional SVR-based foot trajectory training.** Subfigure (a) shows the foot's geometric trajectory. Subfigures (b) and (c) show the horizontal time and phase plane trajectories while subfigures (d) and (e) show the corresponding vertical trajectories. In subfigure (f), the regression-based hyper-surface using our methods is shown with the red line corresponding to the zero contour line. Subfigure (g) and (h) illustrate the horizontal SVR model from different viewpoints. The green data is the contour line with zero Φ value while the red data is the noisy foot trajectory data. In subfigure (h), notice that the fitting captures well the infinity slopes around the zero velocity axis, owed to using implicit regression.

trol approach would be to implement a linear time base controller, e.g. a proportional-derivative-integral controller based on the trajectory error. However, we consider here the addition of a feedforward term and a robust control term to enhance the tracking performance. The hypothesis is therefore that a nonlinear feedforward robust controller plus a linear time-based controller will be more effective than a linear time-based controller alone.

In particular, we consider sliding mode control (SMC) for robustness [72].

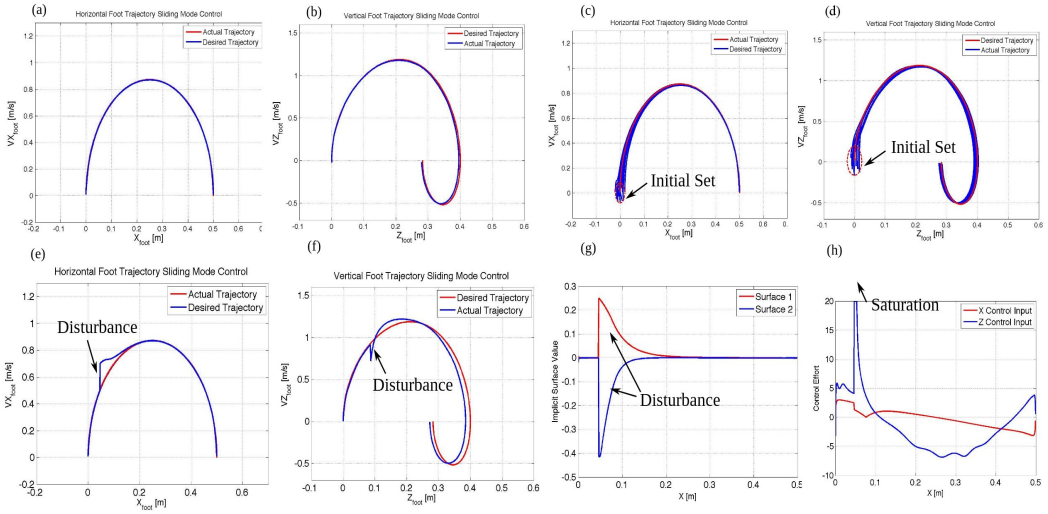


Figure 5.2: **Surface based SMC applied to a trajectory with various initial conditions and external disturbances.** Subfigure (a) and (b) show the SMC method applied to track a nominal trajectory. Subfigures (c) and (d) show the trajectory tracking under various starting conditions. In subfigures (e) and (f), a velocity disturbance is exerted in both the x and z directions. As a result of applying the SMC control law of Equation (5.5), the trajectory converges to the nominal value. Subfigure (g) shows the surface value of the actual trajectory under disturbances. Subfigure (h) shows the SMC control effort, where the blue line is the vertical control input while the red line is the horizontal control.

We implement theory here of an SMC approach for foot swinging during locomotion. Our SMC-based controller is based on the regression model previously described. Consider the implicit surface $s \triangleq \Phi(\mathcal{X})$ and its time derivative

$$\dot{s} = \frac{d\Phi(\mathcal{X})}{dt} = \frac{\partial\Phi(\mathcal{X})}{\partial x}\dot{x} + \frac{\partial\Phi(\mathcal{X})}{\partial\dot{x}}\ddot{x} \tag{5.1}$$

$$= -2\gamma\sum_{i=1}^{l}(\alpha_i - \alpha_i^*)[(x - x_i)\dot{x} + (\dot{x} - \dot{x}_i)\ddot{x}]A_{exp} \tag{5.2}$$

Notice that the state ($\mathcal{X}$) above now corresponds to the foot trajectory and not

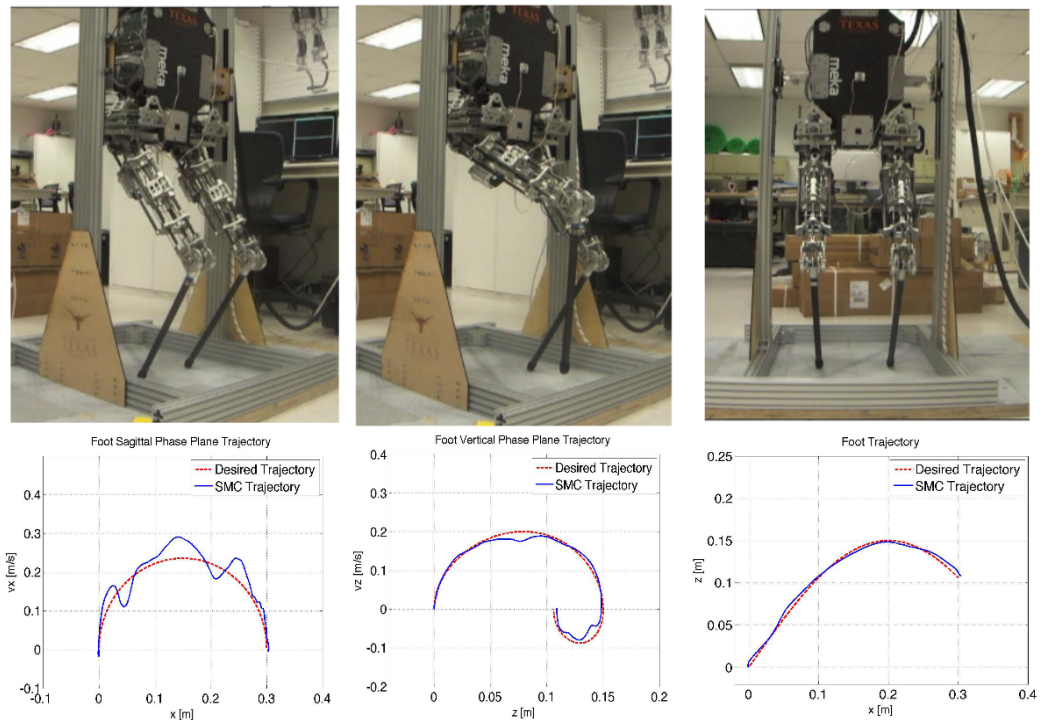


Figure 5.3: **Initial experiments on our Hume bipedal robot.** In this sequence we show the implementation of a swing leg motions described in Fig. 5.1, and based on the implicit regression process with sliding mode control described in Equation (5.6). The experiment shows the leg swinging to a height, as if it was stepping toward a staircase step. The controller is effective on tracking the desired path, with an accuracy of less than 5 mm.

to the robot's center of mass like we had considered before. We leverage sliding mode control theory for asymptotic convergence to the desired trajectory, e.g.

$$\dot{s} = -\eta \cdot \tanh(s) \tag{5.3}$$

This step is potentially a contribution on robust control on itself as sliding mode control normally focuses on convergence to an equilibrium point but not to a trajectory. However, more research to validate this claim will need to be

conducted.

Let us make the following controls observation. One class of controllers that we advocate for humanoid robots is whole-body compliant control [59]. This controllers achieve feedback linearization to render full control of task accelerations, i.e.

$$\ddot{x} = u \tag{5.4}$$

where u is the desired closed-loop control policy. Observing Equations (5.1) and (5.3), we isolate the acceleration term and use it as a feedforward term for the above closed loop dynamics, rendering the robust control law

$$u = \frac{\frac{\eta \cdot \tanh(s)}{2\gamma} - \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)[(x - x_i)\dot{x}]A_{exp}}{\sum_{i=1}^{l}(\alpha_i - \alpha_i^*)(\dot{x} - \dot{x}_i)A_{exp}} \tag{5.5}$$

This law is composed of two parts,

$$u_s \triangleq \frac{\eta \cdot \tanh(s)}{2\gamma \sum_{i=1}^{l}(\alpha_i - \alpha_i^*)(\dot{x} - \dot{x}_i)A_{exp}} \tag{5.6}$$

$$u_{eq} \triangleq \frac{-\sum_{i=1}^{l}(\alpha_i - \alpha_i^*)[(x - x_i)\dot{x}]A_{exp}}{\sum_{i=1}^{l}(\alpha_i - \alpha_i^*)(\dot{x} - \dot{x}_i)A_{exp}}. \tag{5.7}$$

where the term u_s corresponds to the so-called reaching controller, which drives the system dynamics to the desired surface (i.e. the feedback controller), and the second term u_{eq} is the equivalent control, which forces the system dynamics to move along the surface (i.e. the feedforward term).

In Fig. 5.2 we show numerical simulations of the above robust control approach to track desired Phase Plane trajectories of the foot swing. Robustness to initial conditions are shown in Fig. 5.2 (c)-(d) where an initial condition

region is chosen to test the SMC law. To avoid the chattering due to numerical integration, we use variable step integration. We also apply external disturbance via velocity impulse in the Phase Space. The results are shown in Fig. 5.2 (e) - (f).

5.2 Experimental Implementation

Initial experiments on controlling the robot's swinging foot are shown in Figs. 5.3 and 5.4. A software implementation of whole-body compliant control as described in [59] has been developed to run the experiments on the biped. The robot is supported with a boom system that allows for vertical and pitch motion of the torso. Multiple task frames are defined to control the height of the torso, its orientation and the cartesian position of the swing foot. Contact constraints on the support foot are accounted for to solve for the whole-body torques. A geometric trajectory to swing the foot up and forward then back to its original position is implemented and converted to the Phase Plane for robust control. The regression process described in Equation (3.9) and the nonlinear controller described in Equation (5.5) are implemented to track the desired trajectories.

For further validation of the robustness of our controllers, Fig. 5.4 narrates three types of external disturbances applied to the swinging leg. The regression based sliding mode control demonstrates satisfactory robust performance. First, Hume's leg hits a wooden board pushing it away. Second, the leg is hit with a sharp external force. Finally, an elastic band is attached to

the robot's leg which tightens up as the leg moves forward.

The experiment demonstrates that implicit regression techniques are well suited for our robust performance requirements and that the trajectory based sliding mode control strategy indeed works on the real system.

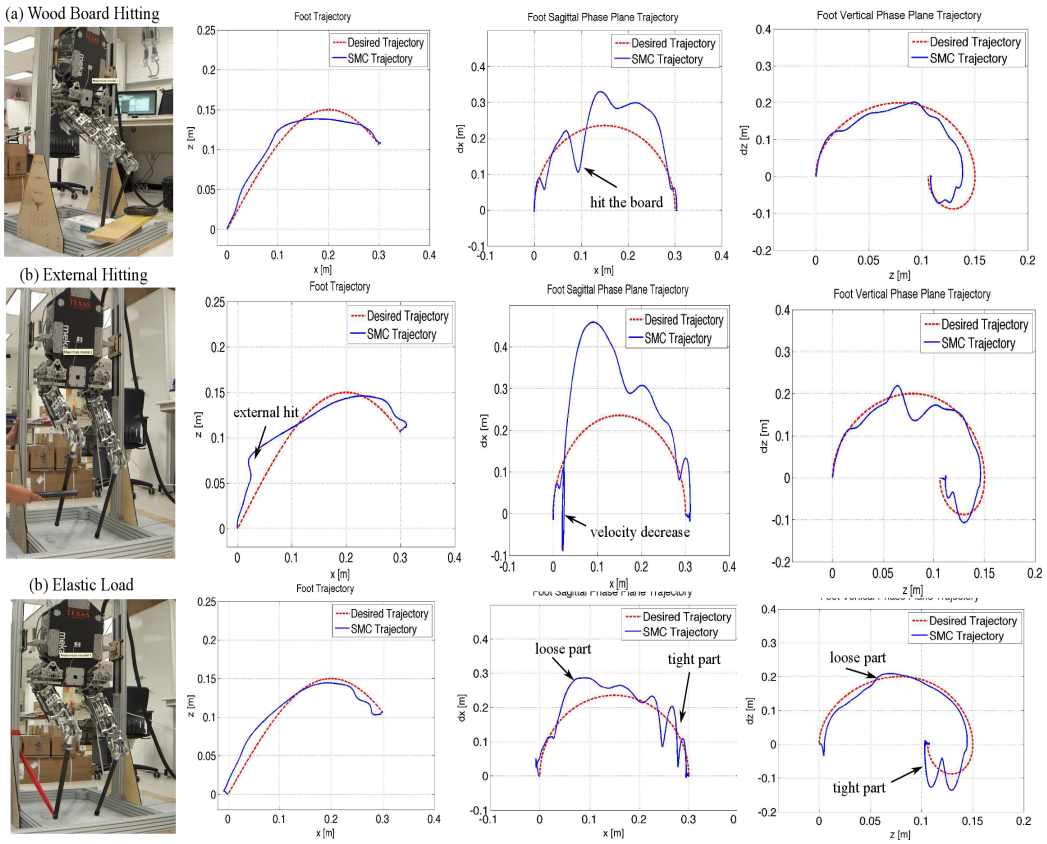


Figure 5.4: **Three experiments under disturbances.** In subfigure (a) the leg hits a wooden board which causes sudden velocity reduction. It follows that the leg pushes the board away to catch up with the nominal path, demonstrating its robustness. Subfigure (b) shows an external force applied to the swing leg, which makes the sagittal velocity to momentarily reverse direction. In subfigure (c), we demonstrate the leg movement when tied up with an elastic band. When the band tightens up, the leg shows some oscillation.

Chapter 6

Conclusions and Future Work

6.1 Thesis Summary

3D legged locomotion can be solved using simple prismatic pendulum models coupled with multi-contact dynamics. According to different CoM surface patterns, two types of inverted pendulum models are elaborated in this thesis and each one processes its own searching algorithm for foot placement. In first model, to reduce the dimensionality of the equations, we propose to define beforehand a 2D non-planar, piecewise linear surface of center of mass geometric behavior. This choice, results in decoupled dynamics of the sagittal and lateral phase behaviors. Meanwhile, to synchronize time, we apply the Newton-Raphson search technique to determine lateral feet locations. Finally, to deal with the non-smooth transitions associated with single contact phases, we introduce multi-contact phases that comply with surface characteristics. As to second model, the CoM is assumed to follow a more general 3D plane and a nested foot placement searching strategy is formulated in a hierarchical way. The walking symmetry and zero CoM lateral velocities at foot apex are guaranteed at the same time.

Data driven strategies require powerful mathematical tools that can solve

the contact transitions of the hybrid system. In particular, regression using implicit functions becomes a necessity to fit complex data sets in the Phase Plane which normally contain infinity slopes and loop behaviors. After designing the hybrid plans, swinging foot trajectories need to be controlled to accurately achieve contact at the desired locations and at the desired time. Time-based linear controllers are a starting point, but a trajectory based feedforward control policy and robust controller can be key to achieve the needed fast response and final accuracy. In this thesis, we have developed a sliding mode control strategy based on Phase Plane plans of the swinging leg and test their effectiveness in simulation and real environment. Two main contributions of our data driven method are therefore on using regression tools to deal with complex Phase Space locomotion plans and to design robust controllers for precise swing foot tracking. In particular, sliding mode control applied to Phase Plane trajectory tracking instead of conventional time-based trajectory tracking might be our another contribution in robust control. However, this claim will need to be further researched.

Overall, we have proposed a methodology that aims to make feasible rough terrain locomotion at human-like speeds. The overall foot placement planner is depicted in Figure 6.1.

6.2 Future Work

In the next phase, we will focus on design of controllers and experimental validation of our algorithms. We will aim at deriving center of mass models

from full joint dynamic simulations or real experimental data. We will also test the locomotion processes in real walking behaviors. To conduct those experiments, a new boom system allowing for sagittal locomotion is ready now. Further along, implementation of 3D locomotion is also on its way. Also, when the reference trajectories are applied to a real robot, modeling errors, sensor disturbances and external perturbations will cause the robot to deviate from planned trajectories. More details about robust controllers will be explored. We also plan to develop whole-body compliant multi-contact controllers [59] to render desired trajectories while adapting to the collisions endured with the terrain. Recently, we have implemented whole-body control algorithms in a mobile manipulator, demonstrating that it is computationally feasible[60].

Another meaningful work is to consider distributed masses and moments of inertia across the robot's body. Prismatic inverted pendulum is a simplified biped model and modeling errors will inevitably be induced. To remedy this deficiency, multi-body dynamics are necessary because of its consideration of distributed leg structure. We can develop a simulated controller and run forward dynamics to estimate the moments induced by the robot. The computed moments can then be utilized to refine the gait trajectories. Additionally, when using point contacts, the multi-contact dynamics have passive modes that have been briefly mentioned. In the future we plan to describe those detailed dynamics and plan the transitions accordingly.

The last but equally important issue is online re-planning strategy based on real-time sensory feedback. When the robot is in presence of disturbance

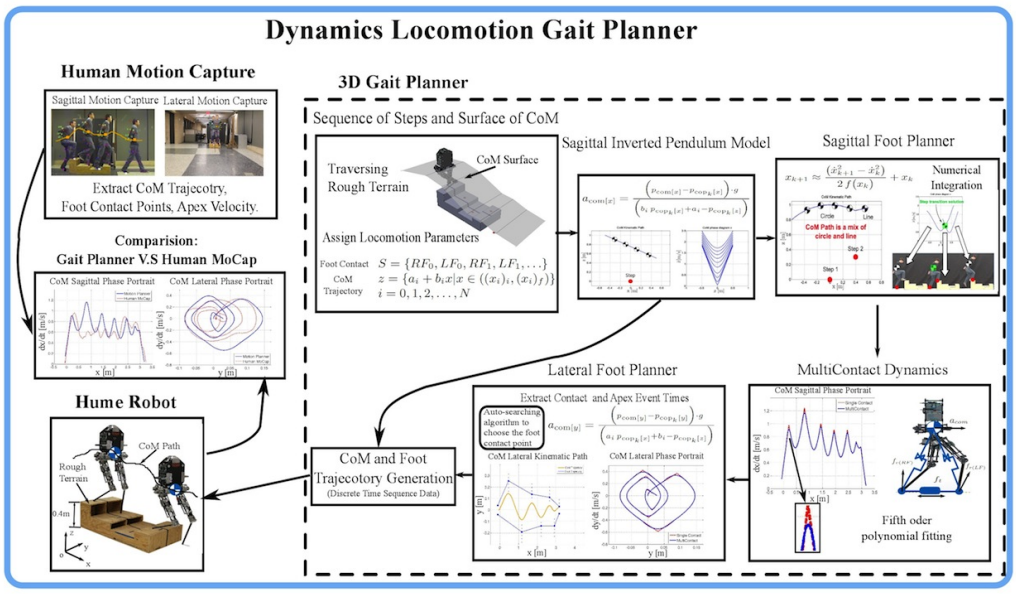


Figure 6.1: **Overall 3D foot placement generator.** This planner diagram depicts the sagittal foot planner (i.e. the step switching policies), the lateral foot planner (i.e. the searching strategy satisfying the timing constraint), and the multi-contact dynamics (i.e. the smoothing of the step transitions). The generated CoM and foot trajectory are extracted to be used as desired trajectory for future control purposes.

or sensory noise, off-line planned trajectories can not be followed as desired. Then, on-line planning [19, 40] is beneficial to achieve accurate yet robust performance. In this case, data driven methods can give a help since it can learn the models based on acquired experimental data and predict what future locomotion behaviors are (for instance, Phase Space contact switching). However, real-time re-planning and control is still challenging due to heavy computational burden from high-dimensional biped states. We potentially resolve manifold learning algorithms to implement dimensionality reduction [68]. That is, we choose certain states (such as, CoM states) as dominant states and use

them to represent the system dynamics [16]. Also, the learned planner should satisfy velocity and acceleration constraints [55]: 1) the position and velocity constraints come from the biped configuration; 2) the acceleration are indirectly constrained by motor torque limitation of the biped. Thus, more results about reachability analysis and safe region should be explored in Phase Space.

Appendices

Appendix A

Planner Validation by Human Walking

To validate our planner, we compare it with data from a motion capture process of a human maneuvering in the rough terrain. It is important to notice that data from the human is not needed for the planner to operate. Therefore it is only used for validation. As shown in Figure A.1, a human subject walks through a wooden rough terrain at speeds varying from 0.5 to 0.9 m/s. The experimental specifications are shown in Table 1. The gait is simultaneously captured by two cameras, one for the sagittal motion and the other one for the lateral motion. We apply scaling algorithms to compensate from perspective variations. We develop a calibration process based on comparing the data from the side and front cameras. Fourteen markers are attached to the body segments and based on the camera information, center of mass behavior is extracted.

In Figure A.1, we compare the results between our algorithms and the data collected from the human. They correlate well with each other except for the first large step. One of the reasons might be that the human relies on ankle behavior to overcome the first obstacle. On the other hand our robot lacks an ankle and therefore needs to gain higher speed to overcome it.

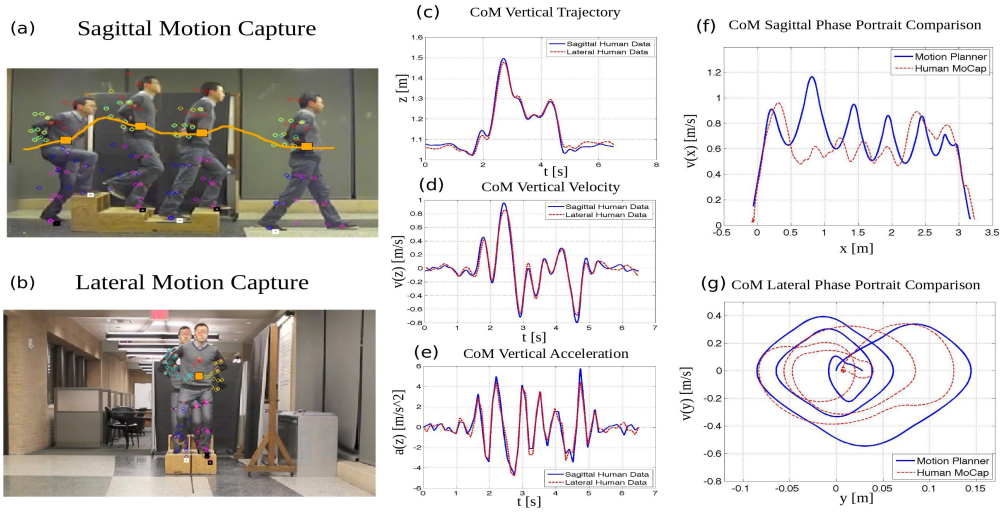


Figure A.1: **Human walking motion capture.** Subfigures (a) and (b) show sagittal and frontal motion data collected using two pocket cameras. In subfigures (c), (d) and (e), we compared vertical human data with the resulting trajectories from the planner. We note the good correlation of the trajectories. In subfigures (f) and (g), we superimposed the sagittal and lateral phase behaviors of the human and the planner.

Table A.1: MoCap Experiment Specification

Parameter	Value	Parameter	Value
Human Weight	70kg	Human Height	183cm
Distance between Wood Board (Right Side) and Front Camera	7.85m	Camera Frame Rate	25Hz
Wood Board Length	1.75m	Wood Board Width	0.7m
Wood Board Left Side Height	0.45m	Walking Steps	7
Walking Speed	0.5m/s	Walking Distance	3.31m

Bibliography

[1] Aaron D Ames. First steps toward underactuated human-inspired bipedal robotic walking. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 1011–1017. IEEE, 2012.

[2] Omür Arslan and Uluç Saranli. Reactive planning and control of planar spring–mass running on rough terrain. *Robotics, IEEE Transactions on*, 28(3):567–579, 2012.

[3] H. Aschemann and D. Schindele. Sliding-mode control of a high-speed linear axis driven by pneumatic muscle actuators. *IEEE Trans. Ind. Electron.*, 55(11):38553864, 2008.

[4] Kostas E. Bekris, B. Y. Chen, A. M. Ladd, E. Plaku, and Lydia E. Kavraki. Multiple query probabilistic roadmap planning using single query planning primitives. In *2003 IEEE/RJS International Conference on Intelligent Robots and Systems (IROS)*, pages 656–661, Las Vegas, NV, October 2003.

[5] Katie Byl and Russ Tedrake. Metastable walking machines. *The International Journal of Robotics Research*, 28(8):1040–1064, 2009.

[6] Katie Byl and Russ Tedrake. Metastable walking machines. *The International Journal of Robotics Research*, 28(8):1040–1064, 2009.

[7] Chih-Chung Chang and Chih-Jen Lin. Libsvm: a library for support vector machines. *ACM Transactions on Intelligent Systems and Technology (TIST)*, 2(3):27, 2011.

[8] Kyong-Sok Chang. *Efficient algorithms for articulated branching mechanisms: dynamic modeling, control, and simulation.* PhD thesis, Stanford University, 2000.

[9] J. Chestnutt, M. Lau, G. Cheung, J. Kuffner, J. Hodgins, and T. Kanade. Footstep planning for the honda asimo humanoid. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 629–634. IEEE, 2005.

[10] Cambrini Chevallereau and Philippe Sardain. Design and actuation optimization of a 4-axes biped robot for walking and running. In *Robotics and Automation, 2000. Proceedings. ICRA'00. IEEE International Conference on*, volume 4, pages 3365–3370. IEEE, 2000.

[11] S. Cotton, I.M.C. Olaru, M. Bellman, T. van der Ven, J. Godowski, and J. Pratt. Fastrunner: A fast, efficient and robust bipedal robot. concept and planar simulation. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 2358–2364. IEEE, 2012.

[12] Hongkai Dai and Russ Tedrake. Optimizing robust limit cycles for legged locomotion on unknown terrain. In *Decision and Control (CDC), 2012 IEEE 51st Annual Conference on*, pages 1207–1213. IEEE, 2012.

[13] Mingcong Deng, Lihua Jiang, and Akira Inoue. Mobile robot path planning by svm and lyapunov function compensation. *Measurement and Control*, 42(8):234–237, 2009.

[14] C. Dune, A. Herdt, O. Stasse, P.B. Wieber, K. Yokoi, and E. Yoshida. Cancelling the sway motion of dynamic walking in visual servoing. In *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*, pages 3175–3180. IEEE, 2010.

[15] J. Engelsberger, C. Ott, M.A. Roa, A. Albu-Schaffer, and G. Hirzinger. Bipedal walking control based on capture point dynamics. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 4420–4427. IEEE, 2011.

[16] Tom Erez and William D Smart. Bipedal walking on rough terrain using manifold control. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 1539–1544. IEEE, 2007.

[17] JP Ferreira, MM Crisóstomo, and AP Coimbra. Adaptive pd controller modeled via support vector regression for a biped robot. *IEEE Transactions on Control Systems Technology*, 21(3):941–949, May 2013.

[18] Gianluca Garofalo, Christian Ott, and A Albu-Schaffer. Walking control of fully actuated robots based on the bipedal slip model. In *Robotics and Automation (ICRA), 2012 IEEE International Conference on*, pages 1456–1463. IEEE, 2012.

[19] Shuzhi Sam Ge, Zhijun Li, and Huayong Yang. Data driven adaptive predictive control for holonomic constrained under-actuated biped robots. *Control Systems Technology, IEEE Transactions on*, 20(3):787–795, 2012.

[20] J.W. Grizzle, C. Chevallereau, A.D. Ames, and R.W. Sinnet. 3d bipedal robotic walking: models, feedback control, and open problems. In *IFAC Symposium on Nonlinear Control Systems*, 2010.

[21] A. Herdt, N. Perrin, P.B. Wieber, et al. Walking without thinking about it. In *IEEE-RSJ International Conference on Intelligent Robots & Systems*, 2010.

[22] T. Hirabayashi, B. Ugurlu, A. Kawamura, and C. Zhu. Yaw moment compensation of biped fast walking using 3d inverted pendulum. In *Advanced Motion Control, 2008. AMC'08. 10th IEEE International Workshop on*, pages 296–300. IEEE, 2008.

[23] R. Hirose and T. Takenaka. Development of the humanoid robot asimo. *Honda R&D Technical Review*, 13(1):1–6, 2001.

[24] J.W. Hurst and A.A. Rizzi. Physically variable compliance in running.

[25] M. Hutter, C.D. Remy, M.A. Hoepflinger, and R. Siegwart. Scarleth: Design and control of a planar running robot. In *Intelligent Robots and Systems (IROS), 2011 IEEE/RSJ International Conference on*, pages 562–567. IEEE, 2011.

[26] M. Hutter, C.D. Remy, MH Hoepflinger, and R. Siegwart. Slip running with an articulated robotic leg. In *International Conference on Intelligent Robots and Systems*, 2010.

[27] B. Jensen, N. Tomatis, L. Mayor, A. Drygajlo, and R. Siegwart. Robots meet humans-interaction in public spaces. *IEEE Trans. on Industrial Electronics*, 52(6):1530–1546, 2005.

[28] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa. Biped walking pattern generation by using preview control of zero-moment point. In *Robotics and Automation, 2003. Proceedings. ICRA'03. IEEE International Conference on*, volume 2, pages 1620–1626. IEEE, 2003.

[29] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Yokoi, and H. Hirukawa. A realtime pattern generator for biped walking. In *Robotics and Automation, 2002. Proceedings. ICRA'02. IEEE International Conference on*, volume 1, pages 31–37. IEEE, 2002.

[30] S. Kajita, F. Kanehiro, K. Kaneko, K. Yokoi, and H. Hirukawa. The 3d linear inverted pendulum mode: A simple modeling for a biped walking pattern generation. In *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, volume 1, pages 239–246. Ieee, 2001.

[31] M. Kalakrishnan, J. Buchli, P. Pastor, M. Mistry, and S. Schaal. Fast, robust quadruped locomotion over challenging terrain. In *Robotics and Au-*

tomation (ICRA), 2010 IEEE International Conference on, pages 2665–2670. IEEE, 2010.

[32] Jeong-Jung Kim, Tae-Yong Choi, and Ju-Jang Lee. Falling avoidance of biped robot using state classification. In *Mechatronics and Automation, 2008. ICMA 2008. IEEE International Conference on*, pages 72–76. IEEE, 2008.

[33] 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. 2011.

[34] J.C. Latombe. *Robot Motion Planning*. Kluwer Academic Publishers, Boston, USA, 1991.

[35] S. M. LaValle. *Planning Algorithms*. Cambridge University Press, Cambridge, U.K., 2006. Available at http://planning.cs.uiuc.edu/.

[36] Steven LaValle. Rapidly-exploring random trees a new tool for path planning. *Report No. TR 98-11, Computer Science Department, Iowa State University.*, 1998.

[37] Steven M LaValle and James J Kuffner. Randomized kinodynamic planning. *The International Journal of Robotics Research*, 20(5):378–400, 2001.

[38] K. Lffler, M. Gienger, and F. Pfeiffer. Sensors and control concept of a biped robot. *IEEE Trans. on Industrial Electronics*, 51(5):972–980, 2004.

[39] Junshui Ma, James Theiler, and Simon Perkins. Accurate on-line support vector regression. *Neural Computation*, 15(11):2683–2703, 2003.

[40] Anirudha Majumdar and Russ Tedrake. Robust online motion planning with regions of finite time invariance. In *Algorithmic Foundations of Robotics X*, pages 543–558. Springer, 2013.

[41] Ian R Manchester, Uwe Mettin, Fumiya Iida, and Russ Tedrake. Stable dynamic walking over uneven terrain. *The International Journal of Robotics Research*, 30(3):265–279, 2011.

[42] P. Michel, J. Chestnut, S. Kagami, K. Nishiwaki, J. Kuffner, and T. Kanade. Gpu-accelerated real-time 3d tracking for humanoid locomotion and stair climbing. In *Intelligent Robots and Systems, 2007. IROS 2007. IEEE/RSJ International Conference on*, pages 463–469. IEEE, 2007.

[43] M. Missura and S. Behnke. Lateral capture steps for bipedal walking. In *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, pages 401–408. IEEE, 2011.

[44] Jun Miura. Support vector path planning. In *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*, pages 2894–2899. IEEE, 2006.

[45] Igor Mordatch, Martin De Lasa, and Aaron Hertzmann. Robust physics-based locomotion using low-dimensional planning. *ACM Transactions on Graphics (TOG)*, 29(4):71, 2010.

[46] Igor Mordatch, Emanuel Todorov, and Zoran Popović. Discovery of complex behaviors through contact-invariant optimization. *ACM Transactions on Graphics (TOG)*, 31(4):43, 2012.

[47] Hirotaka Moriguchi and Hod Lipson. Learning symbolic forward models for robotic motion planning and control. In *Proceedings of European Conference of Artificial Life (ECAL 2011)*, pages 558–564, 2011.

[48] M. Morisawa, S. Kajita, K. Kaneko, K. Harada, F. Kanehiro, K. Fujiwara, and H. Hirukawa. Pattern generation of biped walking constrained on parametric surface. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 2405–2410. IEEE, 2005.

[49] K. Nishiwaki and S. Kagami. Simultaneous planning of com and zmp based on the preview control method for online walking control. In *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, pages 745–751. IEEE, 2011.

[50] E. Ohashi, T. Aiko, T. Tsuji, H. Nishi, and K. Ohnishi. Collision avoidance method of humanoid robot with arm force. *IEEE Trans. on Industrial Electronics*, 45(3):1632–1477, 2007.

[51] J. Pratt, T. Koolen, and etl. De Boer, T. Capturability-based analysis and control of legged locomotion, part 2: Application to m2v2, a lower-body humanoid. 2012.

[52] M. Raibert, K. Blankespoor, G. Nelson, R. Playter, et al. Bigdog, the rough-terrain quadruped robot. In *Proceedings of the 17th World Congress*, pages 10823–10825, 2008.

[53] John Rebula, Fabian Canas, Jerry Pratt, and Ambarish Goswami. Learning capture points for humanoid push recovery. In *Humanoid Robots, 2007 7th IEEE-RAS International Conference on*, pages 65–72. IEEE, 2007.

[54] D. B. Reynolds, D. W. Repperger, C. A. Phillips, and G. Bandry. Modeling the dynamic characteristics of pneumatic muscle. *Ann. Biomed. Eng.*, 32(3):310317, 2003.

[55] Mohammad Jafar Sadigh and Saeed Mansouri. Application of phase-plane method in generating minimum time solution for stable walking of biped robot with specified pattern of motion. *Robotica*, pages 1–15, 2013.

[56] I. Sardellitti, G. Palli, N. Tsagarakis, and D. G. Caldwell. Antagonistically actuated compliant joint: Torque and stiffness control. In *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 13–18, 2010.

[57] L. Sentis. *Synthesis and Control of Whole-Body Behaviors in Humanoid Systems*. PhD thesis, Stanford University, Stanford, USA, 2007.

[58] L. Sentis and B. Fernandez. Perturbation theory to plan dynamic locomotion in very rough terrains. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, September 2011.

[59] L. Sentis, J. Park, and O. Khatib. Compliant control of multi-contact and center of mass behaviors in humanoid robots. *IEEE Transactions on Robotics*, 26(3):483–501, June 2010.

[60] L. Sentis, J. Petersen, and R. Philippsen. Rough terrain whole-body manipulation. In *Autonomous Robots, under review*, 2012.

[61] L. Sentis and M. Slovich. Motion planning of extreme locomotion maneuvers using multi-contact dynamics and numerical integration. In *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, pages 760 –767, oct. 2011.

[62] L. Sentis and M. Slovich. Motion planning of extreme locomotion maneuvers using multi-contact dynamics and numerical integration. In *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, pages 760–767. IEEE, 2011.

[63] M. Slovich. Case studies in multicontact locomotion (advisor: L. sentis). Master thesis degree, The University of Texas at Austin, 2012.

[64] M. Slovich, N. Paine, K. Kemper, A. Metger, A. Edinger, J. Weber, and L. Sentis. Building hume: A bipedal robot for human-centered hyper-agility. In *Dynamic Walking*, 2012.

[65] Michael Slovich. Case studies in multi-contact locomotion. *Master Thesis, UT Austin*, 2012.

[66] Alex J Smola and Bernhard Schölkopf. A tutorial on support vector regression. *Statistics and computing*, 14(3):199–222, 2004.

[67] Koushil Sreenath, Connie R Hill Jr, and Vijay Kumar. A partially observable hybrid system model for bipedal locomotion for adapting to terrain variations. In *Proceedings of the 16th international conference on Hybrid systems: computation and control*, pages 137–142. ACM, 2013.

[68] Mike Stilman, Christopher G Atkeson, James J Kuffner, and Garth Zeglin. Dynamic programming in reduced dimensional spaces: Dynamic planning for robust biped locomotion. In *Robotics and Automation, 2005. ICRA 2005. Proceedings of the 2005 IEEE International Conference on*, pages 2399–2404. IEEE, 2005.

[69] Ioan Alexandru Sucan and Lydia E. Kavraki. Kinodynamic motion planning by interior-exterior cell exploration. In *Algorithmic Foundation of Robotics VIII (Proceedings of Workshop on the Algorithmic Foundations of Robotics)*, volume 57, pages 449–464, Guanajuato, Mexico, 2009. STAR.

[70] R. Tajima, D. Honda, and K. Suga. Fast running experiments involving a humanoid robot. In *Robotics and Automation, 2009. ICRA'09. IEEE International Conference on*, pages 1571–1576. IEEE, 2009.

[71] Russ Tedrake, Ian R Manchester, Mark Tobenkin, and John W Roberts. Lqr-trees: Feedback motion planning via sums-of-squares verification. *The International Journal of Robotics Research*, 29(8):1038–1052, 2010.

[72] Vadim Utkin. Variable structure systems with sliding modes. *Automatic Control, IEEE Transactions on*, 22(2):212–222, 1977.

[73] Vladimir Vapnik. *The nature of statistical learning theory*. springer, 1999.

[74] Liyang Wang, Zhi Liu, Chun Lung Philip Chen, Yun Zhang, Sukhan Lee, and Xin Chen. Energy-efficient svm learning control system for biped walking robots. *IEEE Transactions on Neural Networks and Learning Systems*, 24(5):831–837, May 2013.

[75] Shouyi Wang, Wanpracha Chaovalitwongse, and Robert Babuska. Machine learning algorithms in bipedal robot control. *Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on*, 42(5):728–743, 2012.

[76] Eric R Westervelt, Jessy W Grizzle, Christine Chevallereau, Jun Ho Choi, and Benjamin Morris. *Feedback control of dynamic bipedal robot locomotion*. CRC press Boca Raton, 2007.

[77] K David Young, Vadim I Utkin, and Umit Ozguner. A control engineer's guide to sliding mode control. In *Variable Structure Systems, 1996. VSS'96. Proceedings., 1996 IEEE International Workshop on*, pages 1–14. IEEE, 1996.

[78] Ye Zhao and Luis Sentis. A three dimensional foot placement planner for locomotion in very rough terrains. In *Humanoid Robots (Humanoids), 2011 11th IEEE-RAS International Conference on*, 2012.