Minimalistic Dynamic Climbing

Amir Degani

CMU-RI-TR-10-38

Submitted in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy in Robotics

The Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

November 2010

Thesis Committee:
Matthew T. Mason, Co-Chair
Howie Choset, Co-Chair
Christopher G. Atkeson
Kevin M. Lynch, Northwestern University
Andy Ruina, Cornell University

Abstract

Dynamics in locomotion is highly useful, as can be seen in animals and is becoming apparent in robots. For instance, chimpanzees are dynamic climbers that can reach virtually any part of a tree and even move to neighboring trees, while sloths are quasistatic climbers confined only to a few branches. Although dynamic maneuvers are undoubtedly beneficial, only a few engineered systems use them, most of which locomote horizontally. This is because the design and control are often extremely complicated.

This thesis explores a family of *dynamic climbing* robots which extend robotic dynamic legged locomotion from horizontal motions such as walking, hopping, and running, to vertical motions such as leaping maneuvers. The motion of these dynamic robots resembles the motion of an athlete jumping and climbing inside a chute. Whereas this environment might be an unnavigable obstacle for a slow, quasistatic climber, it is an invaluable source of reaction forces for a dynamic climber. The mechanisms described here achieve dynamic, vertical motions while retaining simplicity in design and control.

The first mechanism called DSAC, for Dynamic Single Actuated Climber, comprises only two links connected by a single oscillating actuator. This simple, open-loop oscillation, propels the robot stably between two vertical walls. By rotating the axis of revolution of the single actuator by 90 degrees, we also developed a simpler robot that can be easily miniaturized and can be used to climb inside tubes. The DTAR, for Dynamic Tube Ascending Robot, uses a single continuously rotating motor, unlike the oscillating DSAC motor. This continuous rotation even further simplifies and enables the miniaturization of the robot to enable robust climbing inside small tubes. The last mechanism explored in this thesis is the ParkourBot, which sacrifices some of the simplicity shown in the first two mechanism in favor of efficiency and more versatile climbing. This mechanism comprises two efficient springy legs connected to a body.

We use this family of dynamic climbers to explore a minimalist approach to locomotion. We first analyze the open-loop stability characteristics of all three mechanisms. We show how an open-loop, sensorless control, such as the fixed oscillation of the DSAC's leg can converge to a stable orbit. We also show that a change in the mechanism's parameters not only changes the stability of the system but also changes the climbing pattern from a symmetric climb to a limping, non-symmetric climb. Corresponding analyses are presented for the DTAR and ParkourBot mechanisms. We finally show how the open-loop behavior can be used to traverse more complex terrains by incrementally adding feedback. We are able to achieve climbing inside a chute with wall width changes without the need for precise and fast sensing and control.

Acknowledgments

Contents

I Open-Loop Climbing

3 DSAC

List of Figures

xiii

List of Tables

Chapter 1

Introduction

Perfection is achieved, not when there is nothing more to add, but when there is nothing left to take away.

— Antoine de Saint-Exupéry

Everything should be made as simple as possible, but not simpler.

— Albert Einstein

Locomotion - the act of moving from place to place, is one of the most basic and important aspects in robotics. This thesis shows how minimalism can be used to design locomoting robots, specifically those that can climb up vertical channels. Minimalism is the attempt to find the simplest mechanism that is capable of performing a given task. Here minimalism is implemented as using fewer motors to achieve stable locomotion without the need of sensing or active control. Similar to the progress in horizontal walking biped robots from sluggish motions and many motors to highly dynamic and minimal control, we set out to explore minimalism in climbing robots. Therefore, the purpose of this thesis is to explore the use of dynamic motions to design minimalistic climbing mechanisms. We present a family of dynamic climbing robots which are inspired by this approach.

1.1 Bipedal locomotion

To give context to climbing locomotion, we summarize the history of bipedal locomotion. Bipedal robotic locomotion has evolved tremendously during the past two decades. However, many of them are still cumbersome, heavy, sluggish and inefficient. Controlling walking robots first started with static walking approaches where the joint angles were carefully programmed to keep the projection of the center of mass (CoM) on the ground inside the foot support area (Kato et al., 1974). This approach suffered from slow speeds and inability to traverse all but flat surfaces. Trying to solve this problem, the zero moment point (ZMP) approach was introduced (Takanishi et al., 1982; Vukobratovic and Borovac, 2004; Vukobratovic and Juricic, 1968). This approach relaxes the constraint of the CoM to be inside the support area by constraining the ZMP, which is the point where there is no moment related to the dynamic reaction force, to be in the support area. A variation of this kind of control is used in the famous Honda Asimo humanoid (Figure 1.1). While this control enables more dynamic and fast walking, the control is based on precise joint-angle control. The main consequence of such a control is that it requires actuators with high precision and frequency response. This results in a heavy mechanism with large motors and sluggish and unnatural movements.

Two minimalistic approaches have followed to allow more dynamic walking and running. These approaches are Raibert's dynamic robots (Raibert, 1986) seen in Figure 1.2 and the passive dynamic walkers (McGeer, 1990b; Collins et al., 2001b) seen in Figure 1.3. Both of these approaches use dynamic motions and relax the need for precise joint angle control. In fact the passive dynamic walking need no control at all! The mechanisms built in Raibert's lab use simple control to stabilize the running direction and speed of single legged hopping robots. The passive dynamic walkers show how a well designed robot on a shallow slope can walk stably in a very human like behavior without the need of any motors. Gravity behaves as the motor for these kinds of machines.
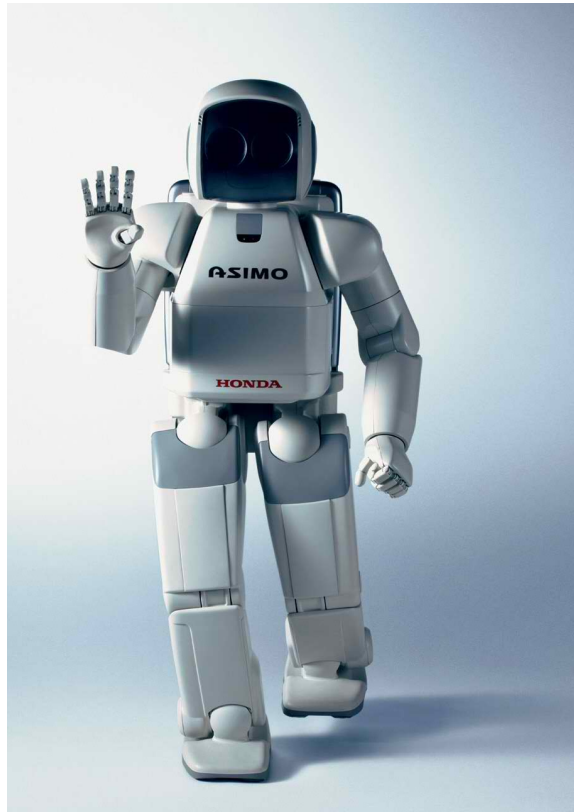
Figure 1.1: The Honda Asimo robot.
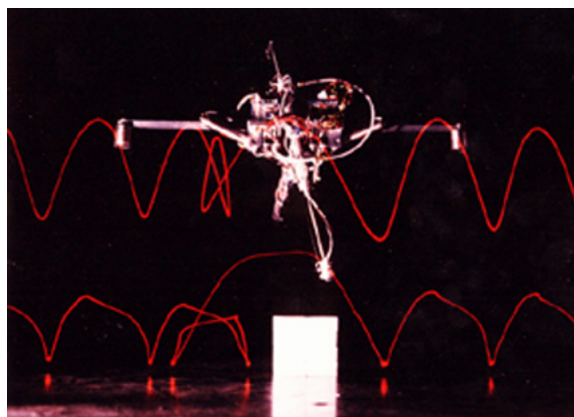
Figure 1.2: One of the hopping robots from Raibert's lab.
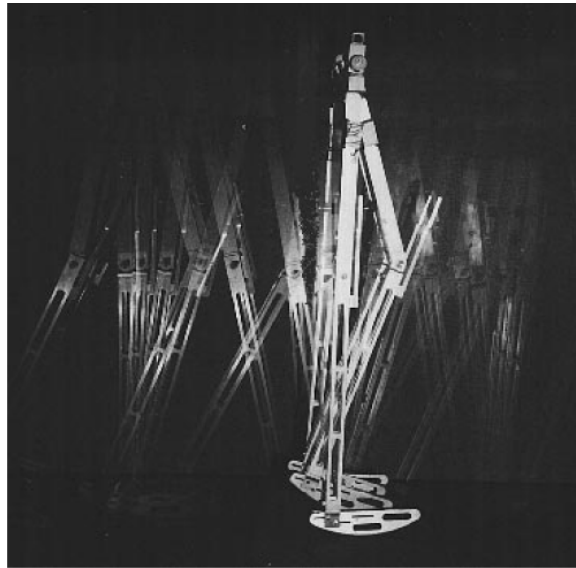
Figure 1.3: Strobe photo of a McGeer-like mechanism walking down a shallow ramp in Cornell's Biorobotics and Locomotion Lab. Reprinted with permission from Andy Ruina. Photo credit to Rudra Pratap.

1.2 Climbing locomotion

We try to take the same approach of using dynamic motions to the climbing task. Climbing is not a trivial task because the mechanism is moving against gravity, hence the mechanism must consist of at least one motor (or another form of energy). The question we asked ourselves is, is it possible to design a mechanism that climbs with a single motor?

Not unlike today's bipedal robot, most of the climbing robots use heavy design and slow motions. In many cases, the quasistatic and ZMP-like motion of the climbing robots not only result in sluggish motions but also in many motors. We next show how the use of dynamic motions can be beneficial in decreasing the number of necessary motions.

1.3 Motivation: Why use dynamics?

Humans and even more so animals use dynamic motions in everyday tasks. Such motions are running, jumping over obstacles, throwing objects, and climbing. Other than speed, what are the advantages of dynamic motions over quasistatic motions? There are two. First, being able to overcome obstacles which are impassible while moving slowly. As an example, say a human rock climber tries to hold onto a distant handhold but cannot reach it. One strategy, albeit dangerous, is to leap upwards to try to grab onto the handhold. This is one way to imagine how a dynamic, leaping movement can help increase reachability. Figure 1.4 depicts another example of a human using dynamic motions to climb inside a chute.

The second reason for using dynamic motions is illustrated by comparing two multi-linked systems. These systems have rigid links, connected serially by revolute actuators. We ask "how many links and actuators are needed to climb up a simple channel as shown in Figure 1.5?" The four link mechanism (1.5(a)) must brace itself with the bottom two links while moving the top two links to reposition its footholds. While trying to brace the top two links, the bottom ones must change configuration, resulting in an eminent slip. On the other hand, the five link mechanism (1.5(b)) is able to brace itself with its upper links while enabling the bottom set to reposition themselves. Figure 1.5(c) compares this five link mechanism (left) with our two link dynamic climber. The mechanism on the left can only move quasistatically, while the one on the right
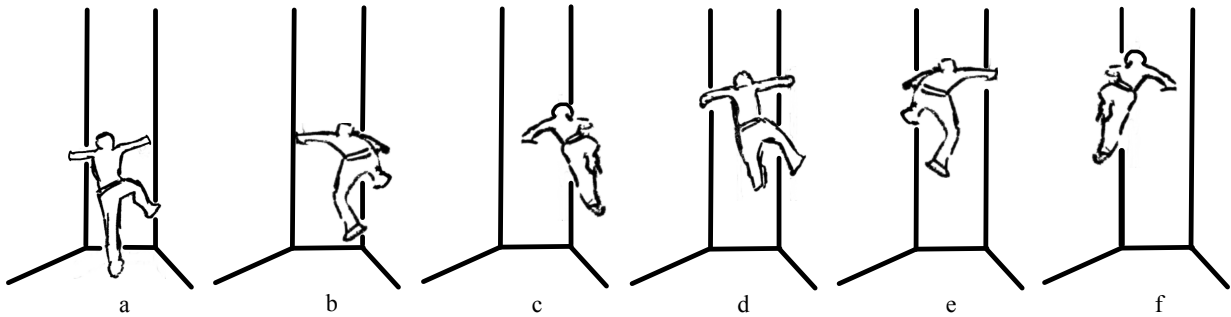
Figure 1.4: Human climbing between two vertical walls.
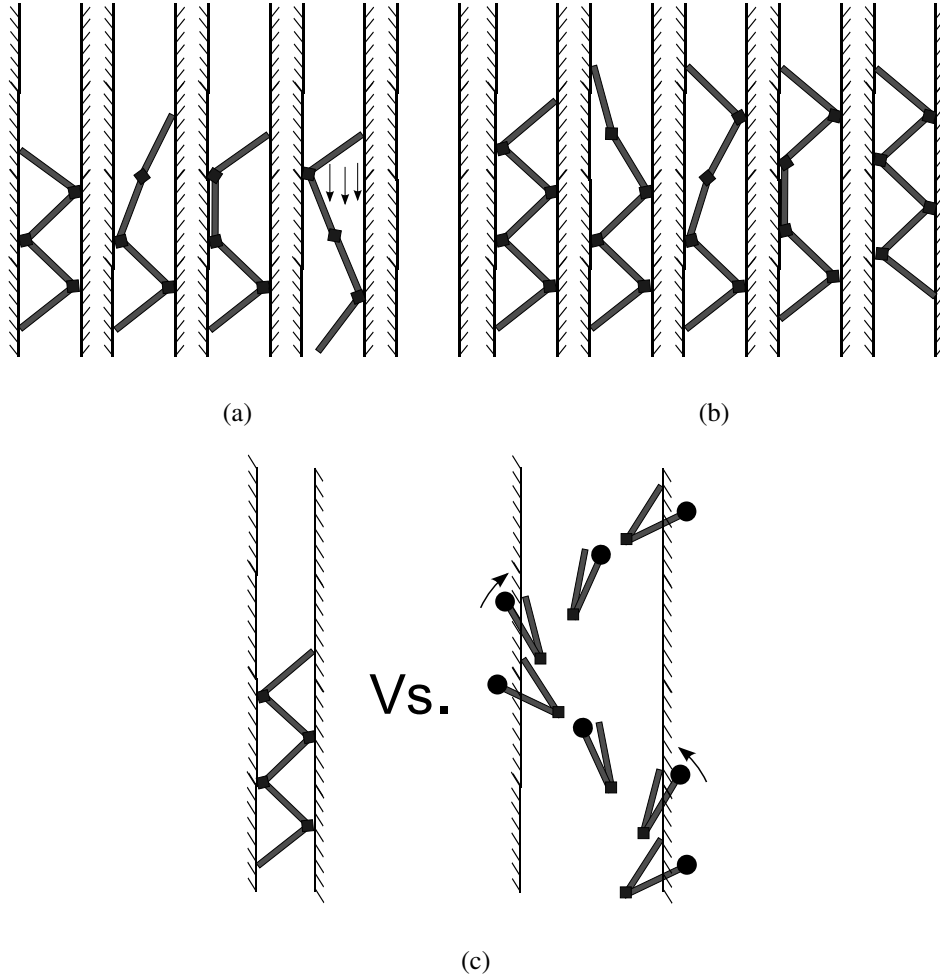
Figure 1.5: Quasistatic multi-linked mechanism climbing inside a chute. (a) An unsuccessful climb of a four linked mechanism. (b) A successful climb of a five linked mechanism.(c) Comparing the five link snake (left) to the two link dynamic climber we present in this thesis.

can employ dynamic strategies such as leaping maneuvers. This example does not prove, but exemplifies how the use of dynamic motions can sometimes decrease the number of necessary motors.
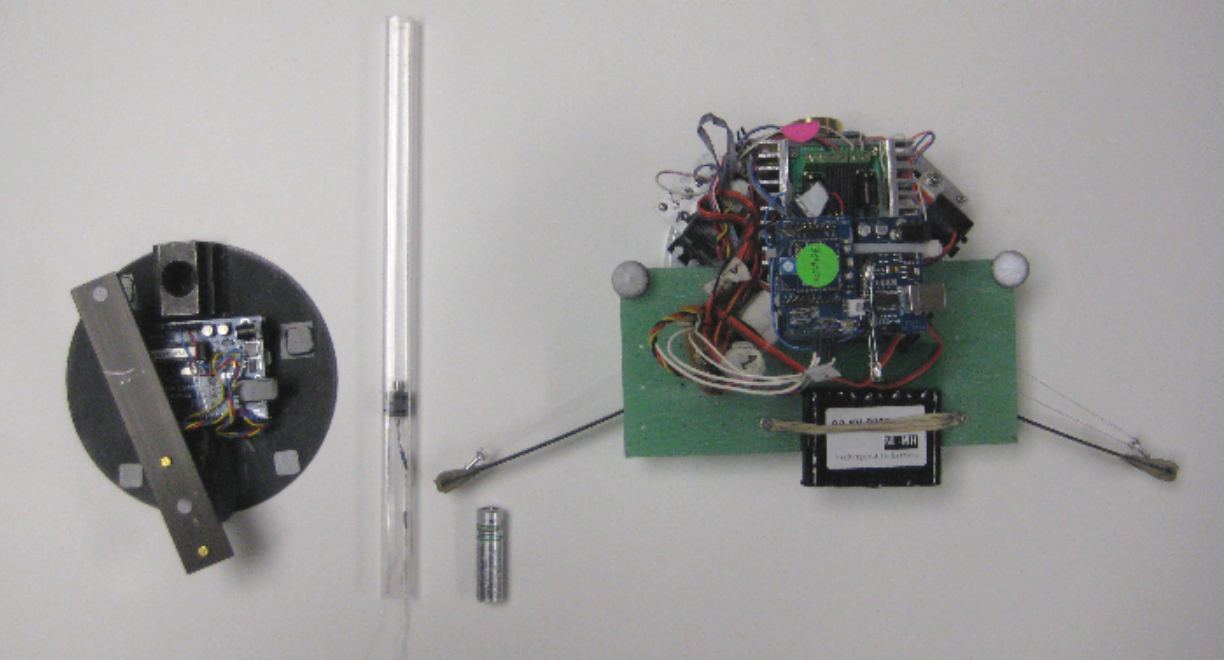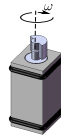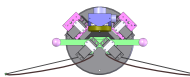
Figure 1.6: The dynamic climbing robots. DSAC (left), DTAR (middle), and ParkourBot (right). Standard AA battery placed in the middle for reference.

1.4   Systems Description

This thesis describes a family of dynamic climbing mechanisms, shown in Figure 1.6. They all use dynamic motions to climb up vertical walls with an open-loop, self stabilizing motion. The first is the DSAC, for Dynamic Single Actuated Climber. The DSAC comprises a single actuated joint connecting two links. By using dynamic motions this mechanism climbs up a chute between two parallel walls. The second is a miniaturized extension to the DSAC called DTAR, for Dynamic Tube Ascending Robot. The third dynamic climbing robot is the ParkourBot. This mechanism is more complex, however it is more efficient and can be controlled in a more precise fashion.

Table 1.1 summarizes the three systems analyzed in this thesis. Common to these three systems is dynamic motions. Moreover, these three systems are inherently stable without feedback control.

Table 1.1: Systems description

Name	Description	Section	Image
DSAC	comprises a single oscillating motor connecting two links.	Open-loop: Chap. 3 Closed-loop: Sec. 6.3	
DTAR	a tube climbing robot similar to the DSAC. Comprises only a rigid body, two O-rings and a motor rotating an eccentric mass.	Open-loop: Chap. 4	
ParkourBot	comprises two springy legs connected to a body. During flight, the robot stores elastic energy in its springy legs and automatically releases the energy to "kick off" the wall during touch down.	Open-loop: Chap. 5 Closed-loop: Sec. 6.4	
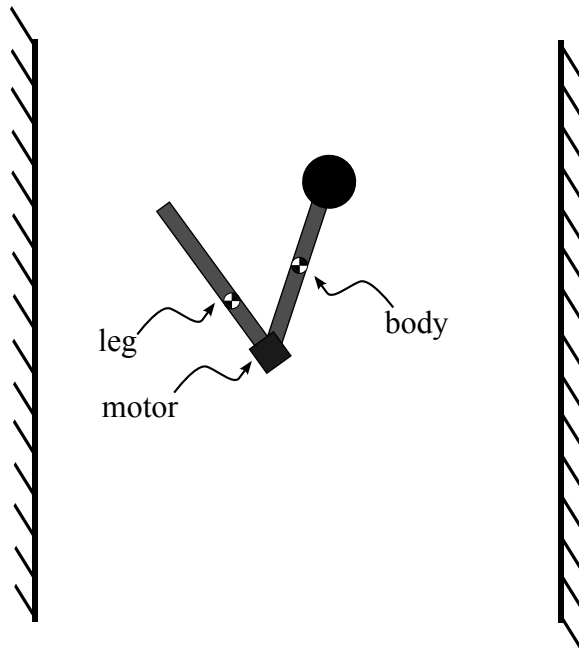
Figure 1.7: Schematics of the two link mechanism in flight between two parallel walls.

1.4.1 DSAC

The DSAC mechanism is planar and consists of two links; the first is the main body and the second is the leg. The leg, which contacts the wall, is connected to the main body through an actuated revolute joint (Figure 1.7). We show that even when the motor outputs a simple symmetric oscillation, such as a sinusoid, the mechanism, under some choices of parameters, will climb stably. Variation of the mechanism's parameters alters the behavior of the mechanism significantly. We have identified two "typical" climbing motions: single contact and double contact. The former only contacts the distal part of the leg, and the latter also contacts the proximal of the leg (see Figures 1.8 and 1.9, respectively).

Single contact climbing is advantageous in that it can climb wider gaps. However, in general, the frequencies required from the motor are larger than for double contact climbing. On the other hand, double contact climbing is restricted to narrower gaps. Double contact climbing is less dynamic since most of the climbing motion follows from rotations around the points of
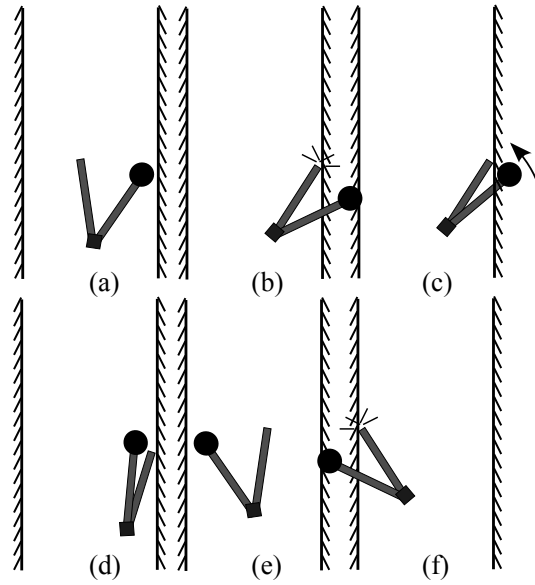
Figure 1.8: Example of a single contact climbing motion of the DSAC. Only the distal end of the leg hits the walls. This climbing gait can climb wide gaps, but in general the motor has to oscillate quickly.

contact and not while flying between the walls. We will mostly focus on the more dynamic single contact climbing motion.

To help understand how the DSAC climbs, it is helpful to decompose the motion from one wall to the other into three phases; flight, impact and stance (see Figure 1.8(a,b,c), respectively). The flight phase is just a simple continuous motion without any external forces applied to the body, other than gravity. The impact phase can be regarded as an instantaneous phase where the configuration does not change but the velocities do change instantaneously. The stance phase is when the pendulum-like body swings toward the counter wall and the leg is in contact with the wall.

It is valuable to examine each phase in order to understand how the mechanism is capable of climbing stably without any sort of control loop. The analysis section will show that during the impact phase, the leg loses most of its angular velocity, which acts as a reset function to reduce the accumulated perturbation after each impact. The stance phase has two "tasks". First, the body

10
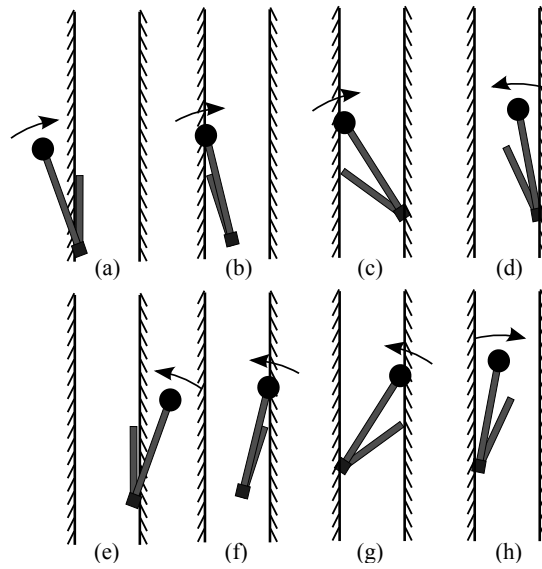
Figure 1.9: Example of a double contact climbing motion. This is the less dynamic gait where both the distal and the proximal ends of the leg hit the walls. In order to climb the leg has to be similar in length to the wall width. This is a very robust climbing gait and requires lower motor speeds than the single point contact climb.

swings toward the counter wall which makes the leg "stick" to the wall. Second, while swinging the body, the mechanism absorbs kinetic energy which gives it the kick during the transition to flight phase. The flight phase is when the mechanism gains height and changes configuration that enable a continuous climbing motion. This motion is further explained more formally in the next chapters. One might imagine this motion as the human from the cartoon in Figure 1.4. The kickoff as in Figure 1.4(c) is equivalent to the stance phase. This phase imparts energy into the system until it transitions to flight phase. Figures 1.4(d,e) resemble the flight and impact phases, respectively.

1.4.2   DTAR

The DTAR, for Dynamic Tube Ascending Robot, is an extension of the DSAC (see Figure 1.10). Instead of oscillating the leg of the DSAC around an axis perpendicular to the direction of climbing, the DTAR continuously rotates around the axis in the climbing direction. This resolves the problem of using an oscillatory motor and enables miniaturization of the robot. To implement this motion we use a simple "pager motor" which continuously rotates an eccentric mass. This eccentric mass is equivalent to the leg of the DSAC. In fact, by looking at a projection of the mass on a plane, this mass follows a sinusoid similar to the DSAC leg angle. The DTAR prototype comprises a small rigid body (approximately 1cm by 1cm by 1cm), two O-rings and a motor rotating the eccentric mass. The mass ratio and geometry of this prototype make the mechanism climb in a gait similar to the one shown in Figure 1.9. This "double contact climbing" is very robust. As will be shown in Chapter 4, we analyze two important conditions that must hold for stable climbing and determine the mechanism's parameters that ensure a stable gait sequence. We further approximate the climbing rate of DTAR.
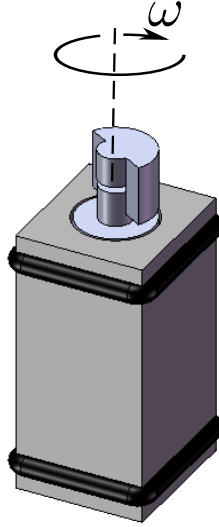
Figure 1.10: Cad model of the DTAR mechanism. The DTAR prototype comprises a small rigid body (approximately 1cm by 1cm by 1cm), two O-rings and a motor rotating the eccentric mass.

1.4.3 ParkourBot

The ParkourBot, shown in Figure 1.11 is an efficient, two-legged, dynamic climbing robot. The robot comprises two springy legs connected to a body similar to the leg design of the BowLeg hopping robot (Brown and Zeglin, 1998; Zeglin and Brown, 1998). Leg angle and spring tension are independently controlled. The robot climbs between two parallel walls by leaping from one wall to the other. During flight, the robot stores elastic energy in its springy legs and automatically releases the energy to "kick off" the wall during touch down. Chapter 5 elaborates on the mechanical design of the ParkourBot. We use a simplified spring-loaded inverted pendulum (SLIP) model to simulate the ParkourBot motion and stability. Finally, we detail experimental results, from open-loop climbing motions to closed-loop stabilization of climbing height.

13
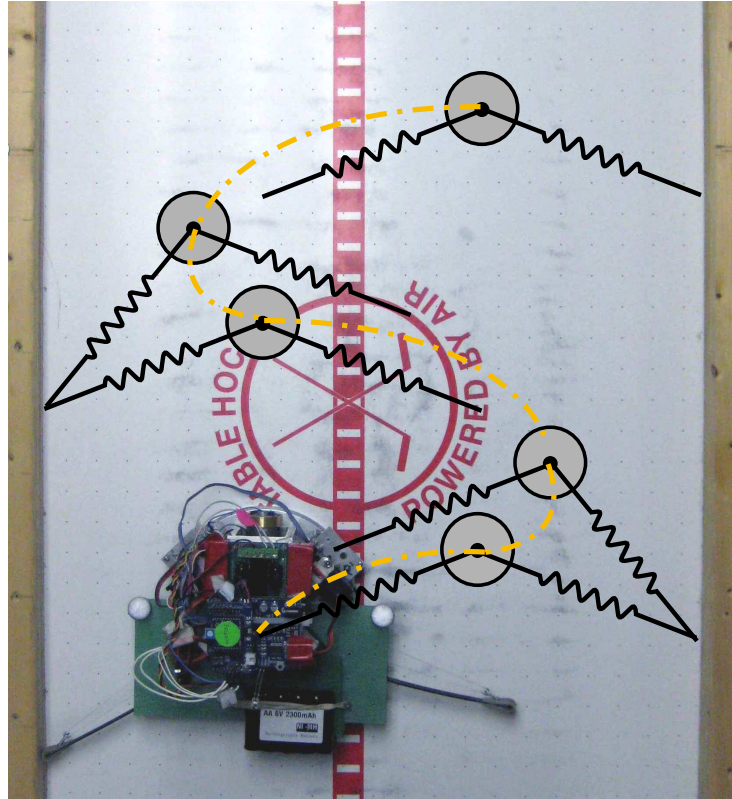
Figure 1.11: The ParkourBot overlayed with the simplified SLIP model climbing motion.

1.5 Thesis outline

The outline of this document is as follows. Chapter 1.7 describes related work, including climbing mechanisms, walking machines, hopping robots, planning for dynamic systems, and analysis of dynamic systems. Chapter 2 summarizes the relevant mathematical preliminaries used in succeeding chapters.

This thesis includes two parts. Part I analyzes the open-loop characteristics of the three mechanisms and Part II describes an algorithm to enable climbing in a more complex terrain with minimal addition of sensing.

1.5.1 Part I: Open-loop climbing

The DSAC (Chapter 3), DTAR (Chapter 4) and the ParkourBot (Chapter 5) all exhibit stable climbing motions without need of external sensing and feedback. Part I explores these open-loop mechanisms. Simple models of each mechanism will be presented, together with derivations of the equations of motion. This will be followed by open-loop stability which includes the local stability analysis and the basin of attraction approximation. The DTAR analysis explores the range of parameters that will allow stable climbing inside tubes. Proof-of-concept experiments are given for all three mechanisms.

The DSAC and ParkourBot exhibit interesting nonlinear phenomena, including period doubling bifurcation and quasi-periodic motions which will be investigated in depth. We further show how in some cases a non-symmetric "limping" period-2 motion can be more efficient and even more robust than the symmetric motion.

1.5.2 Part II: Closed-loop climbing

The knowledge from the open-loop analysis is used to address more complex environments with just a small addition of feedback and sensory information. Chapter 6 in Part II presents an

algorithm that uses basins of attraction approximation to find a graph of possible transitions between controls and terrains. The final goal is to climb between walls of changing width using this algorithm.

1.6 Publication Note

Analysis and experiments of the DSAC mechanism was first published in (Degani et al., 2007). Much of Chapter 3 has appeared in (Degani et al., 2010a). The DTAR mechanism was first analyzed in a video submission (Degani et al., 2010b). Chapter 4 will appear in (Degani et al., 2010d), and Chapter 5 which was submitted to ICRA 2011 is joint work with Ben Brown, Kevin Lynch and Siyuan Feng.

1.7 Related Research

This mechanism is unique but its underlying mechanisms draw from many areas including minimalist manipulation, walking and hopping robots, open-loop controlled robots and planning for dynamical systems. These are briefly summarized in this chapter.

1.7.1 Minimalism

In the context of this work, the minimalist approach is the attempt to find the simplest mechanism that is capable of performing a given task. Simplicity of a system can be defined in different ways. In general one tries to minimize the amount of sensory input, actuation or computation. Previous minimalism works have dealt with manipulation and locomotion. Canny and Goldberg (1995) examined how a simple system comprising of a parallel-jaw gripper and an optical beam sensor, together with geometric planning and sensing algorithms is capable of recognizing and orienting a broad class of industrial parts. Erdmann and Mason built a tray-tilting system which can orient a part in a random initial configuration in the tray without sensing (Erdmann and Mason, 1988).

Lynch and Mason (1999) planned and controlled dynamic nonprehensile manipulation. They used a one degree of freedom arm to perform dynamic tasks such as snatching an object from a table, rolling the object on the arm, and throwing and catching. A good example of minimalism in the locomotion task is the passive dynamic walkers described below (McGeer, 1990a,b; Garcia et al., 1998).

The mechanisms described in this thesis extend the minimalism in locomotion from horizontal motions to vertical, climbing motions. The mechanism can perform a climbing task, albeit a simple one, without sensing and control, with a single actuator and a simple mechanical design.

1.7.2 Walking robots

McGeer, who initiated the work on passive dynamic walking (McGeer, 1990a,b) showed that a properly designed walking machine can walk down a gentle slope without any active control or energy input, other than potential energy from the slope. The mass and link length parameters can be chosen so that the natural dynamics of the walker enters a stable limit cycle from a basin-of-attraction of initial conditions. This principle has been used in the design of passive walkers with counter-swinging arms (Collins et al., 2001a) and low-power walkers capable of walking over flat ground (Collins et al., 2005). We use a similar tactic in our mechanism but instead of using gravity as a "dumb" actuator, we use a fixed symmetric oscillation.

1.7.3 Hopping robots

Dynamic climbing is in many senses similar to dynamically locomoting robots, in particular hopping, passive dynamic walking, and running robots. The work of Raibert was particularly influential, as it demonstrated that simple control laws could be used to stabilize hopping and control the running speed and direction of 2D and 3D single-leg hoppers (Raibert and Brown, 1984; Raibert et al., 1984). The single-leg systems also serve as models for runners with multiple legs (Raibert et al., 1986; Raibert, 1986). This work inspired detailed analysis of the nonlinear

dynamics of a hopping robot (Vakakis et al., 1991; Koditschek and Bühler, 1991) and gymnastic maneuvers in both simulation (Berkemeier and Fearing, 1998; Mombaur et al., 2005a) and experiments (Hodgins and Raibert, 1990). To improve the energy efficiency of a hopping robot, Brown and Zeglin introduced the BowLeg hopper, which can traverse a series of stepping stones (Brown and Zeglin, 1998; Zeglin and Brown, 1998) using a highly efficient bow-like spring. The BowLeg concept is the basis of our ParkourBot mechanism discussed in Chapters 5,6.

1.7.4 Simplified models

To facilitate analysis and control design of running and walking robots, it is convenient to develop simplified models which nonetheless retain the essential character of the original physical system. Two examples are the spring-loaded inverted pendulum (SLIP) model of running robots (Raibert, 1986; Blickhan, 1989; Blickhan and Full, 1993) and the "simplest walking model" (Garcia et al., 1998). Such models can be used to extract important relationships between design and control parameters and performance. For example, Kuo used the simplest walking model to demonstrate that applying an impulse at toe-off is a more energy-efficient way to inject energy into a walker than applying a torque to the stance leg (Kuo, 2002). In the current work, we develop a simplified model of the ParkourBot to analyze the open-loop dynamic stability in the chute-climbing task. The chute-climbing task may be viewed as "vertical running," in that our goal is to stabilize a desired limit cycle motion, as in running robots.

1.7.5 Climbing mechanisms

1.7.5.1 Quasistatic climbers

One aspect of this work that differs from the work described above is that locomotion occurs largely in the vertical direction. While a number of robots have been designed for climbing locomotion, they are mostly quasistatic. The Alicia3 robot climbs walls by using pneumatic adhesion at one or more of three "cups" connected by two links (Longo and Muscato, 2006).

(a) adhesive: Suction and Magnets   (b) spines   (c) brute force gripper   (d) grasping/bracing
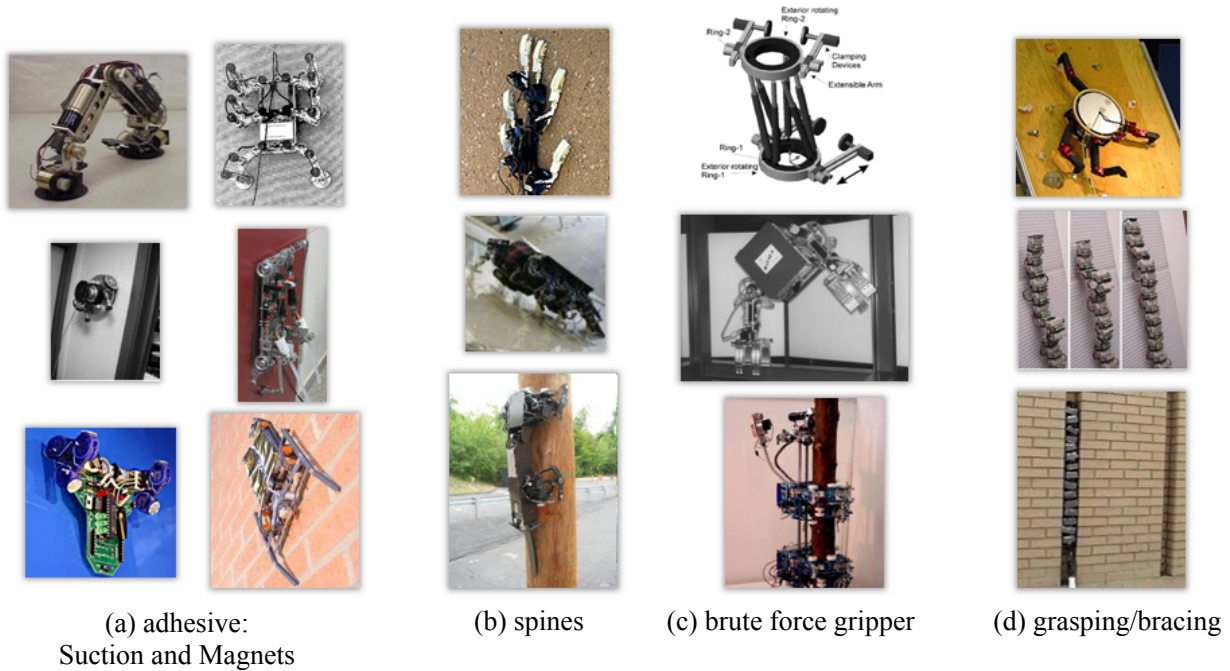
Figure 1.12: A few examples of quasistatic climbing mechanisms divided into four groups depending on the "attachment mechanism". (a) Climbing mechanisms using suction or magnets such as the Stickybot and Wallbot. (b) Mechanisms using spines such as SpinybotII and RiSE. (c) Mechanism using brute force gripping. (d) Mechanisms bracing between opposing wall such as the LEMUR robot.

The climbing robots of Shapiro et al. (2005) and Greenfield et al. (2005) climb by kinematic or quasistatic bracing between opposing walls. Bretl (2006) and Bevly et al. (2000) both use foothold based climbing strategies. Specifically, the four-limbed free-climbing LEMUR robot goes up climbing walls by choosing a sequence of handholds/footholds, as well as motions to those footholds, that keep the robot in static equilibrium at all times (Bretl, 2006). Gecko-inspired directional dry adhesives allow Stickybot and Waalbot to climb vertical, smooth surfaces such as glass (Kim et al., 2007; Murphy and Sitti, 2007), and the RiSE and SpinybotII robots climb soft or rough walls using spined feet to catch on asperities in the wall (Autumn et al., 2005; Kim et al., 2005). Figure 1.12 presents some of these quasistatic climbing mechanisms.

1.7.5.2 Dynamic climbers

Unlike the quasistatic climber, only a few mechanisms have been proposed to achieve a vertical climbing task using dynamic motions. Clark et al. (Clark et al., 2006, 2007) are in the process of making the clawed RISE and SpinyBot robots more dynamic. They analyzed and designed a cockroach inspired dynamic climbing robot which resembles a biologically based template for dynamic vertical climbing. Their robot comprises a main rigid body with two linearly moving hands with springs. A few differences set the DSAC apart from their dynamic climber. First, their mechanism is more complex in design since it uses two motors, energy storing springs, and a crank mechanism. Second, its climbing motion is similar to brachiating, flightless motion. During all times one arm is fixed to the ground. Lastly, in contrast to the family of climbing robots in this thesis, the cockroach inspired robot does not use reaction forces from walls but rather uses spines to attach itself to a carpet covered wall. Similarly the ROCR robot, Jensen-Segal et al. (2008) uses spines to attach to a carpeted wall and a single actuator rocking a pendulum to swing up and climb.

1.7.5.3 Meso-scaled climbing robots

The DTAR mechanism analyzed in Chapter 4 is a meso-scaled tube climbing robot. Two classes of meso-scaled tube climbing mechanisms in the order of 1-100mm have been previously proposed: quasistatic and dynamic. Quasistatic climbers use slow motions to move from one stable pose to the next such as the inchworm crawlers (e.g., Menciassi et al. (2006); Wang et al. (2008); Kassim et al. (2006); Slatkin and Burdick (1995)). Dynamic mechanisms are mostly vibratory system such as (Gmiterko et al., 2002; Mištinas and Spruogis, 2002; Salomon et al., 2008). These systems use canted bristles to produce anisotropic friction. Energy transferred into the system produces a motion in the direction of lowest friction. Power to these mechanisms is generated either by an external power source (e.g., Salomon et al. (2008)) or an internal vibrating mechanism. This anisotropic friction produces the asymmetry which is crucial to locomote in a

preferred direction. In the DTAR mechanism the asymmetry is not achieved through friction but from locating the moving mass above the most distal contact point. Not relying on anisotropic friction enables the operator to change the direction of motion, and to safely withdraw the mechanism when power is shut off. A mechanism shown in (Milano et al., 2009) is similar to the DTAR in design but locomotes on flexible guide wires.

1.7.6 Open-loop control

The classic control method of locomoting robots is feedback control, where the loop is closed in real-time using fast sensors and complicated feedback algorithms. For these reasons high onboard computation is required and large amount sensory information. Ringrose (1997) and Mombaur et al. (2005a) have developed open-loop controls for dynamic hoppers. Ringrose (1997) used large circular feet to stabilize a one legged hopper. Mombaur and colleagues (Mombaur et al., 2005a,b) showed an approach which is similar to our work. In their work one- and two-legged robots exhibit self-stabilizing running motions without closed-loop feedback. Two optimization loops are used: an outer loop for finding stable motions by changing robot model parameters, such as length and masses, and an inner loop which searches for an optimal control by minimizing the control inputs under the robot constraints. We intend to adopt similar optimization techniques to find an optimal design and control of the DSAC for fast climbing.

Seyfarth, Geyer and Herr show how in bipedal running, an open-loop strategy of retracting the swing leg between the apex of flight phase and before it impacts the ground not only allows the foot velocity to better match the ground before impact, but also tends to stabilize the running speeds (Seyfarth et al., 2003). In many ways this is similar to the approach of open-loop vertical juggling where negative acceleration of the batter at the nominal impact time tends to stabilize the bouncing height of the ball (Schaal and Atkeson, 1993).

1.7.7 Planning for dynamic systems

The broader goal of this thesis, other than investigating the open-loop, simple, climbing mechanisms, is to use minimal information to traverse harder terrain. This requires some control and planning. Similar to the control of a hopping robot, or in our case a climbing robot, several researchers used simple planning and control to perform juggling. As examples, Aboaf et al. (1989) used learning control, Lynch and Black (2001) used control based on gradient descent about a nominal batting trajectory, Ronsse et al. (2006) used control with minimum feedback (impact time only), and Bühler and Koditschek (1990), Rizzi and Koditschek (1992) introduced the "mirror law" control to juggle one or two objects in 2D and 3D.

Chapter 6 discusses the use of the open-loop stable climbing as primitives to traverse between more complex terrain. Each one of these open-loop climbing motions can be thought as a funnel converging initial state condition toward the stable orbital attractor. Mason (1985) used the term funnel as an analogy for eliminating uncertainty in manipulation by using purely mechanical means without the needs of sensors. Lozano-Perez et al. (1984) introduced the notion of pre-image backchaining for fine manipulation. They partitioned the state space into cells with different local controllers which when deployed correctly the goal can be backchained into a large set of initial conditions. These works focused on quasistatic problems with Coulomb reaction forces restricted to piecewise constant-velocity control actions. Burridge et al. (1999) verified theoretically the validity of the backchaining approach to non constant control policies on Newtonian dynamics models. Weingarten et al. later used sequential composition for switching between control policies for the Rhex (Weingarten et al., 2004) legged system. Recent advancements in numerical approximation of these funnels, which are in essence the basin of attraction or the Lyapunov function of a system, led to new and efficient algorithms in planning for nonlinear systems (Tedrake et al., 2010). Recently Gregg et al. (2010) used geometric reduction-based controls to find a set of asymptotically stable "primitives" for a 3-D bipedal robot, each corresponding to walking along a nominal arc of constant curvature for a fixed number of steps. They

then composed these primitives (or funnels) in a controlled manner to produce a stable walking path. Unlike these works where only a controller changes, our approach will also include terrain change as a distinct "funnel". This enables the machine to traverse varying terrain with a simple global planner.

Chapter 2

Background and Preliminaries

This chapter provides some background and preliminaries which will help to understand the following chapters. While it covers a wide array of topics this chapter is not intended to be a comprehensive explanation of all the concepts used in this thesis. The chapter begins with the general formulation of the equations of motion of an articulated body together with added external forces. Nondimensionalizing equations of motion will finish this part of the chapter. Nonlinear dynamical systems definition and analysis methods conclude the chapter, including orbital stability, Poincaré maps, and bifurcations.

2.1 General Equations of Motion

2.1.1 Lagrangian formulation of equations of motion

The equations of motion differentially connect the input torques Υ and the resulting motion of the mechanism in state space. There are many ways to generate dynamic equations of motion of a mechanism. In this thesis, the Lagrangian analysis is used to derive the equations of motion.

We use the classical definition of generalized coordinates, $q = (q_1, \ldots, q_n) \in Q$, of the mechanism as a minimal set of n independent coordinates which specifies the posture of the

robot, where n is the number of degrees of freedom of the mechanism. This generalized coordinate vector contains joint angles and the cartesian location of one reference point on the robot relative to an inertial frame. The velocity vector $\dot{q} = (\dot{q}_1, \ldots, \dot{q}_n) \in T_q Q$ comprises both angular and linear velocities. The entire state, including configuration and velocities, is $z = (q, \dot{q}) \in TQ$.

In order to derive the equations of motion, the Lagrangian $L : TQ \to \mathbb{R}$ is defined for a mechanical system as the difference between the kinetic and potential energy of the system.

$$L(q, \dot{q}) = T(q, \dot{q}) - V(q),$$

where T is the kinetic energy and V is the potential energy of the system written in generalized coordinates.

Definition 2.1 (Lagrange's equations (Murray et al., 1994; Greenwood, 1997)). The equations of motion for a mechanical system with generalized coordinates $q \in \mathbb{R}^m$ and Lagrangian L are given by

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = \Upsilon_i \quad i = 1, \ldots, m, \tag{2.1}$$

where Υ_i is the external force acting on the ith generalized coordinate. These are the generalized forces.

If the ith joint is passive the generalized force vanishes, i.e., $\Upsilon_i = 0$. It is sometimes convenient to rewrite Eq. 2.1 in matrix form

$$M(q)\ddot{q} + h(q, \dot{q}) = \Upsilon \tag{2.2}$$

where for an n degree of freedom robot, $M(q) \in \mathbb{R}^{n \times n}$ is the positive definite symmetric mass matrix, $h(q, \dot{q}) \in \mathbb{R}^{n \times 1}$ is a vector summarizing the influence of Coriolis, centrifugal, and gravitational forces, and $\Upsilon \in \mathbb{R}^{n \times 1}$ is a vector containing the generalized forces.

2.1.2 Lagrange's equations with contact forces

Like many other articulated locomoting mechanisms, our mechanism uses the contact with the walls to locomote. When the mechanism is in contact with an obstacle these contact forces

26

can be easily incorporated into the Lagrange equations by formulating corresponding contact constraints. These constraints come from the assumption that no sliding occurs at the contact and therefore no work is done by the contact constraint forces. Note that the constraint itself is the fact that the velocity of the contact point itself is zero, not the work at that point. The contact constraints can be written as

$$A(q)\dot{q} = 0 \quad A(q) \in \mathbb{R}^{k \times n}. \tag{2.3}$$

where the constraint matrix A represents a set of k velocity constraints. In our kind of system, n is the number of degrees of freedom of the system and k is the number of constraint forces. When a system having a single unilateral constraint such as our mechanism hitting the wall, $k = 2$ which will include normal and tangential contact forces. This constraint matrix can be derived as $A(q) = \frac{\partial P(q)}{\partial q} \in \mathbb{R}^{2 \times 3}$, where $P(q) = (P_x \quad P_y)^T$ is the point of contact with the walls. The dynamics of the system can now be written as

$$\frac{d}{dt} \frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = \Upsilon_i - A^T(q)\lambda_{ext} \tag{2.4}$$

where λ_{ext} (the Lagrange multipliers) are the contact forces with the obstacle. Eq. 2.4 can be written in matrix form as

$$M(q)\ddot{q} + h(q, \dot{q}) = \Upsilon - A^T(q)\lambda_{ext} \tag{2.5}$$

By differentiating 2.3 and solving for $\ddot{q}$ from Eq. 2.5, the Lagrange multipliers are obtained

$$\lambda_{ext}(q, \dot{q}) = \left(A(q)M(q)^{-1}A(q)^T\right)^{-1} \left(\dot{A}(q)\dot{q} - A(q)M(q)^{-1}h(q, \dot{q})\right). \tag{2.6}$$

For a more careful derivation of these equations with constraints see the book by Murray et al. (1994).

2.1.3 State space representation

In many cases it is useful to transform the n second order differential equations into $2n$ first order differential equations which is also called state space representation. By first defining the state vector as $z = (q, \dot{q})^T$, the $2n$ first order differential equations are written as

$$\dot{z} = \begin{bmatrix} \dot{z}_1 \\ \dot{z}_2 \end{bmatrix} = \begin{bmatrix} \dot{q} \\ \ddot{q} \end{bmatrix} = \begin{bmatrix} \dot{q} \\ M(q)^{-1} \left(\Upsilon - A^T(q)\lambda_{ext} - h(q,\dot{q}) \right) \end{bmatrix}$$
$$= \begin{bmatrix} z_2 \\ M(z_1)^{-1} \left(\Upsilon - A^T(z_1)\lambda_{ext} - h(z) \right) \end{bmatrix} \triangleq f(z, \Upsilon) \tag{2.7}$$

2.1.4 Autonomous system vs. Non-Autonomous Systems

So far, the equations of motion were written as a system of equations

$$\dot{z} = f(z, \mu), \tag{2.8}$$

where $z = (q, \dot{q})$ is the state of the system. The parameters of the mechanism and the environment are denoted as μ. These parameters include the link lengths, masses, and the distance between walls. A system of ordinary differential equations is *autonomous* when it does not depend on time (or another independent variable). In contrast, a system is *non-autonomous* when it does depend on time. In 2.8, since the right hand side does not include time it is an autonomous system. A non-autonomous system is of the form

$$\dot{z} = f(z, t, \mu); \quad (z, t) \in \mathbb{R}^n \times \mathbb{R}, \tag{2.9}$$

A nth-order time-periodic non-autonomous system with period T (i.e., $f(\cdot, t) = f(\cdot, t + T)$) can always be converted into an $n+1^{th}$-order autonomous system of differential equations (Guckenheimer and Holmes, 1983; Parker and Chua, 1989)

$$\dot{z} = f(z, \tau, \mu),$$
$$\dot{\tau} = 1; \quad (z, \tau) \in \mathbb{R}^n \times \mathbb{S}^1, \tag{2.10}$$

where τ is the new state component representing time. The phase space has now been transformed into the manifold $\mathbb{R}^n \times \mathbb{S}^1$ (cylinder), where $\mathbb{S}^1 = \mathbb{R} \pmod{T}$ is the periodicity of the system. Note that for a periodic forcing of the shape $A\sin(\omega t)$, as in our system, another natural choice of the new state component can be $\tau = \omega t$, and therefore $\dot{\tau} = \omega$.

2.1.5   Nondimensionalizing Differential Equations

We show that with a sinusoidal motor trajectory, and with the correct choice of parameters, the mechanism climbs stably. In order to reduce the dimension of the parameter space, we use a method called nondimensionalizing, or normalization. This method described in Appendix A can reduce the number of parameters by up to the number of fundamental units. In our system the number of parameters that can be reduced is three: mass, length, time. Instead of using the entire set of parameters, the idea of nondimensionalizing is to deal with ratios of these parameters. This will convert a system of differential equations into unitless (dimensionless) parameters. This method not only reduces the parameter space, but can also give intuitive and physically meaningful ratios of the parameters. For example, instead of looking at two masses of the system, the nondimensionalized parameter might be the ratio between the masses, or the ratio between one mass and the total mass of the system. Appendix A describes in detail the motivation and work flow of nondimensionalizing differential equations.

2.2   Nonlinear analysis

So far we have described the general formulation of the equations of motion for an articulated robot such as our climbing mechanism. The rest of this chapter deals with the methods used to analyze nonlinear dynamical systems. The section begins with an overview of orbital stability analysis and then describes more accurately how to use the Poincaré map to simplify the analysis of systems with limit cycles and finally describes a few nonlinear phenomena and the means to interpret them.

2.2.1   Orbital stability - the Poincaré map

As mentioned previously, our system exhibits periodic motions due to the forced periodic constraint ($\phi(t)$). Moreover, the system is a *hybrid* system, one that cannot be described as a single

continuous flow but only as a collection of continuous flows with discrete changes during the transitions. In our system, these discrete changes occur while impacting the walls. Due to these facts, a useful tool to analyze stability is the Poincaré map (Guckenheimer and Holmes, 1983). This tool converts the study of the hybrid periodic flow of our mechanism into a nonlinear discrete mapping on a lower dimensional space. By looking at the crossing of the flow with a cross section one can now analyze this discrete system instead of the more complicated hybrid flow. Period-1 motions, i.e., climbing motion which returns to its initial state after one period, will correspond to a single fixed point on the Poincaré section. Period-k motions, i.e., flow that returns to the same state after k periods, will correspond to k points on this section. The Poincaré map defined in this work, maps one state of the climbing robot, just after leaving the wall, to the state where the robot leaves the next wall. This is done by solving the equations of motion of the flight, impact, and stance phases numerically.

To find fixed points, we use the multidimensional Newton-Raphson numerical root finding method. To analyze the orbital stability, the Poincaré map is linearized around these fixed points. This linearization is the Jacobian at the fixed points. We find both stable and unstable fixed points. If the eigenvalues of this Jacobian are inside the unit circle then a perturbation from a limit cycle will converge to the unperturbed state, and this fixed point is said to be stable.

We now turn to more accurate definitions. As in Eq. 2.8, an autonomous differential equation can be described as

$$\dot{z} = f(z), \tag{2.11}$$

where $z \in \mathbb{R}^n$ is the state of the system and $f : \mathbb{R}^n \to \mathbb{R}^n$ is a Lipschitz-continuous vector field. Thus, there exists exactly one solution for every initial condition $z(t_0) = z_0$. The solution is denoted by the trajectory $z(t)$ or by the flow $\Phi_t(z_0)$. The flow $\Phi_t(z_0)$ assigns a trajectory $z(t)$ to every initial value z_0.

Definition 2.2 (Periodic Solution, Periodic Orbit). A solution $\Phi_t(z_0)$ of Eq. 2.11 is a periodic solution with period length $T > 0$ if $\Phi_{T+t}(z_0) = \Phi_t(z_0)$ holds for all times $t \in \mathbb{R}$.

The choice of z_0 is not unique; any point of the periodic solution is a valid starting value for an autonomous system.

Definition 2.3 (Poincaré map)**.** Consider an n-dimensional system as in 2.8. Let γ be a periodic orbit of some flow Φ_t in $\mathbb{R}^n$ arising from the nonlinear vector field $f(z)$. Let $\Sigma \subset \mathbb{R}^n$ be an $n-1$ dimensional cross section. The cross section Σ need not be planar, however, it must be transverse to the flow, i.e., all trajectories starting on Σ flow through it, not parallel to it. Denote the unique point where γ intersects Σ by p. Then the Poincaré map $\mathbf{P} : U \to \Sigma$ is defined in a neighborhood $U \subset \Sigma$ of p as

$$\mathbf{P}(q) = \Phi_\tau(q).$$

where $\tau = \tau(q)$ is the time taken for the orbit $\Phi_t(q)$ based at q to first return to Σ. If the mapping has a fixed point z^*, then $\mathbf{P}(z^*) = z^*$ (see Figure 2.1). If multiple mapping is required for a state to return to itself, i.e.,

$$\mathbf{P}^k(z^*) = z^*,$$

the system has a period-k cycle, i.e., after k cycles that state maps back to the initial state. On the Poincaré section a period-k cycle will correspond to k points.

There are a few common types of Poincaré maps which differ by the chosen section. The two typical ones which are used in this analysis are the stroboscopic Poincaré map and the impact Poincaré map. The former is the more common map which takes the Poincaré section every equal time interval. The latter, which is commonly used in analyzing hybrid robotic systems, uses an event such as a leg of a robot hitting the floor as the Poincaré section.

2.2.2 Stability definition

One of the most important and interesting questions asked when analyzing a dynamical system is whether the system is stable or not. Since our system is periodic, the orbital stability must be analyzed. As mentioned previously one can convert our system from a continuous or a hybrid system into a discrete system by using the Poincaré map method. Therefore instead of analyzing
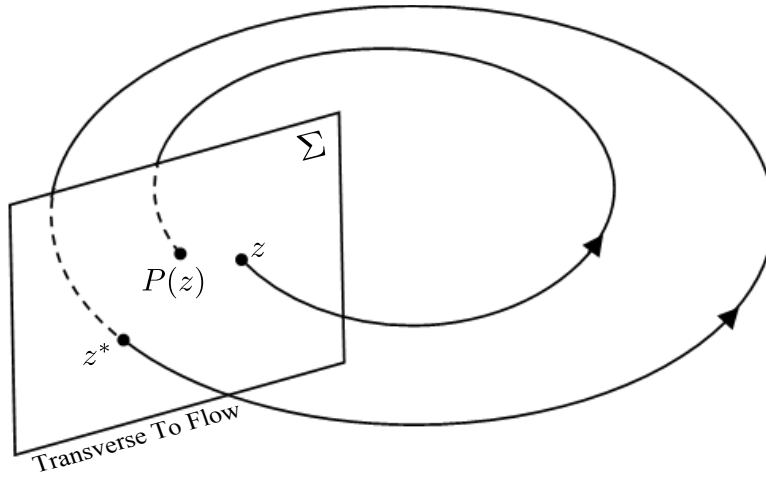
Figure 2.1: Poincaré map.

the stability of an orbit, the stability of a fixed point on the Poincaré section is analyzed. This fixed point corresponds to a closed orbit in the full state space. Defining stability on the full state space of a periodic system is problematic since two identical flows with phase different will not be considered stable. However, when using the Poincaré method these two solutions will be both converge to the same point and therefore stable. There are a few different forms of stability.

Definition 2.4 (Stable (Lyapunov Stable) Fixed Point). A fixed point z^* of $f(z)$ is called stable if for any given neighborhood $U(z^*)$ there exists another neighborhood $V(z^*) \subseteq U(z^*)$ such that any solution starting in $V(z^*)$ remains in $U(z^*)$ for all $t \geq 0$.

Loosely speaking being Lyapunov stable means stability in the weak sense that trajectories starting nearby a limit cycle will remain nearby for all time. Asymptotic stability which is defined next also adds the constraint that at steady-state the flow is attracted back to the original limit cycle.

Definition 2.5 (Asymptotically Stable Fixed Point). A fixed point z^* of $f(z)$ is called asymptotically stable if it is stable and if there is a neighborhood $U(z^*)$ such that

$$\lim_{t \to \infty} |\Phi(t, z) - z^*| = 0 \quad \text{for all } z \in U(z^*)$$

In order to find if a fixed point of the Poincaré map is asymptotically stable Floquet analysis

32

is used. Floquet analysis looks at the eigenvalues of the linearized map around the fixed point to determine its stability.

Theorem 2.1 (Characteristic (Floquet) Multipliers). *Let z^* be a fixed point of the Poincaré map* **P**. *The map* **P** *is n-dimensional for a non-autonomous systems and $n - 1$-dimensional for an autonomous systems. The local behavior of the map near z^* is determined by linearizing the map at z^*. The linear map, called the Jacobian Matrix or the monodromy matrix, is*

$$\delta z_{k+1} = D\mathbf{P}(z^*)\delta z_k,$$

where δz_k and δz_{k+1} are a perturbation from the fixed point at iteration k and $k+1$, respectively. The eigenvalues of the Jacobian $D\mathbf{P}(z^*)$ are the characteristic multipliers of the periodic solution. These characteristic multipliers govern the evolution of perturbation δz_0 around the fixed point. (Parker and Chua, 1989; Nayfeh and Balachandran, 1995).

- *If all of the characteristic multipliers are within the unit circle, then the corresponding fixed point is asymptotically stable. Hence, the associated periodic orbit is asymptotically stable and is an attracting limit cycle. This fixed point is called an attractor.*

- *If all of the characteristic multipliers are outside the unit circle, the corresponding fixed point is unstable. Therefore, the associated periodic orbit is an unstable limit cycle. This fixed point is called a repellor.*

- *If some, but not all, of the characteristic multipliers are outside the unit circle, the corresponding fixed point is a saddle. Hence, the associated periodic orbit is an unstable limit cycle of the saddle type. See Figure 2.2.*

Note, this classification scheme remains valid as long as none of the characteristic multipliers lies on the unit circle. A fixed point with no characteristic multipliers on the unit circle is called hyperbolic. The stability of a non-hyperbolic fixed point cannot be determined from the characteristic multipliers alone unless one characteristic multiplier has magnitude greater than one and another characteristic multiplier has a magnitude less than one, in which case the fixed point

33
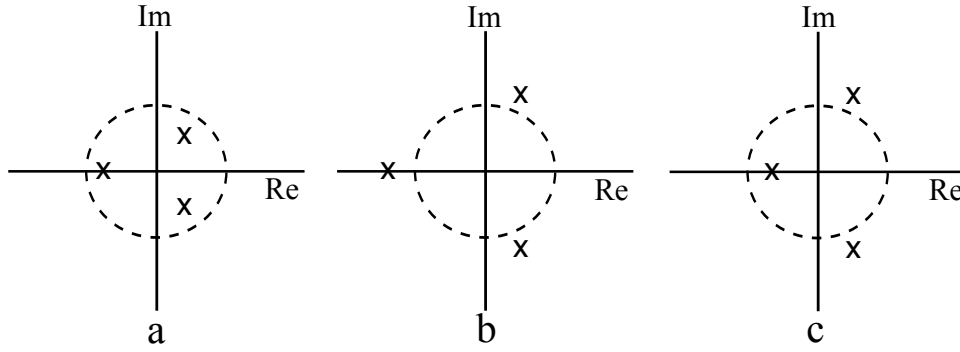
Figure 2.2: The position in the complex plane of the characteristic multipliers at a hyperbolic fixed point determines the stability of the fixed point. (a) asymptotically stable (b) unstable (c) non-stable, saddle type.

is non-stable. To characterize non-hyperbolic fixed points, one must investigate the higher order nonlinear terms.

Proof. If one looks at the map of a perturbed state $z^* + \delta z_{k+1} = \mathbf{P}(z^* + \delta z_k)$, expanding the result in a Taylor series about z^* and retaining the linear terms, one obtains

$$z^* + \delta z_{k+1} = \mathbf{P}(z^* + \delta z_k) \approx \mathbf{P}(z^*) + D\mathbf{P}\delta z_k.$$

Therefore a perturbation from the fixed point decays if all the eigenvalues of DP are less than one. Hence, the system is locally stable around this linearization. The smaller the eigenvalue, the faster this perturbation decays. See (Nayfeh and Balachandran, 1995) or (Parker and Chua, 1989) for the full proof. □

2.2.3 Bifurcations of nonlinear systems

Definition 2.6 (Bifurcation). Once again consider an nth-order system

$$\dot{z} = f(z, \mu)$$

with a parameter $\mu \in \mathbb{R}$. As μ changes, the steady-state solution of the system also changes. If a small change in μ causes a steady-state solution to undergo a qualitative change it is called a
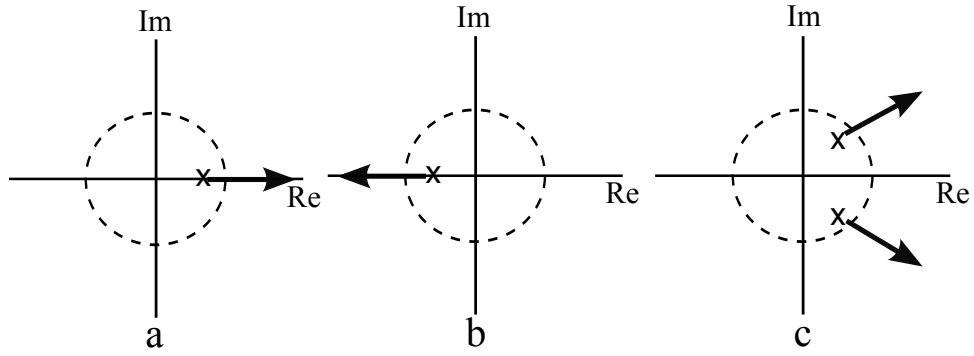
Figure 2.3: Scenarios depicting how the Floquet multipliers leave the unit circle for different local bifurcations: (a) transcritical, symmetry breaking, and cyclic-fold bifurcations; (b) period doubling bifurcation; and (c) secondary Hopf or Neimark bifurcation. (Nayfeh and Balachandran, 1995)

bifurcation and the value at which a bifurcation occurs is called a bifurcation value (or point). Note that typically a small change in μ produces small quantitative changes in a steady-state solution. For instance, perturbing μ could change the position of a steady-state solution slightly, and if the steady-state solution is not an equilibrium point, its shape or size could also change.

The bifurcations can be analyzed using the characteristic multipliers defined in Definition 2.1. Three typical bifurcations related to the characteristic multipliers leaving the unit circle are shown in Figure 2.3. We will use these analysis method to plot the characteristic multiplier locus and check for bifurcation while changing a parameter continuously.

2.2.4 Types of dynamic motions

There are three classic types of dynamic motions which are relevant to our climbing mechanism:

- equilibrium (fixed point);

- periodic motion or a limit cycle;

- quasiperiodic motion which contains a finite number of incommensurable frequencies [1].

[1]Frequencies are incommensurable if their ratio cannot be expressed as a ratio of whole numbers.

These motions are called attractors, because if some form of damping is present the transients decay and the system is "attracted" to one of the above three states.

The fourth kind of motion is called Chaos, which is associated with a "strange attractor". We will not go into details about this kind of motion, however we will try to define it in a simple manner and later show how to analyze and find chaotic motions. The main engineering motivation in searching for a chaotic region is to try to avoid these motions.

Definition 2.7 (Chaotic Motion). Chaos is aperiodic long-term behavior in a deterministic system that exhibits sensitive dependence on initial conditions

Aperiodic long-term behavior means that the system does not reach a steady state solution of one of the above attractor solutions: equilibrium, periodic motion or quasiperiodic motion. A system is **deterministic** when the later states of the system follow from the earlier ones. In dynamical systems this implies that the system has no random or noisy inputs or parameters. **Sensitive** dependance to initial condition occurs when two very close initial conditions diverge exponentially from each other. In the next section we will briefly show a few methods to analyze periodic solutions and search for chaotic regions.

2.2.5 Methods of analyzing nonlinear systems

2.2.5.1 The Poincaré map

The most popular method to analyze periodic, forced systems is the Poincaré map which was previously discussed. By observing the crossing of the system through the Poincaré section, one can easily distinguish between different motions.

- A k-periodic motion maps to k points on the Poincaré section, i.e., period-1 motion maps to a single point on the Poincaré section, period-2 motion maps to two points, etc.

- A quasiperiodic motion which contains a finite number of incommensurable frequencies traces a continuous closed curve on the Poincaré map since it does not converge to a single point.

- If the Poincaré map does not consist of either a finite set of points or a closed curve, the motion **may** be chaotic [2].

2.2.5.2 Lyapunov exponents

One of the important characteristics of chaos is sensitive dependance on initial conditions. The Lyapunov exponents can reveal if indeed there is an exponential relationship between the flow of two very close initial conditions. In general, for an n-dimensional dynamical system, there are n Lyapunov exponents. To check for sensitivity of initial conditions, only the largest Lyapunov exponents is of interest. The method for finding this largest Lyapunov exponent is very similar to finding the Lyapunov exponent of a one-dimensional map which is explained next.

Definition 2.8 (Lyapunov Exponents for 1-D maps). Assume P is a Poincaré map of a 1-D system. Let z_0 and $z_0 + \Delta z_0$ be two nearby initial points on the flow, not necessarily in steady state. After one iteration of a map the points are separated by

$$\Delta z_1 = P(z_0 + \Delta z_0) - P(z_0) \simeq \Delta z_0 P'(z_0)$$

where $P' = dP/dz$. The local Lyapunov exponent λ at z_0 is

$$\lambda = ln|\frac{\Delta z_1}{\Delta z_0}| \simeq ln|P'(z_0)|$$

To obtain the global Lyapunov exponent, an average of the local Lyapunov exponent over many iterations must be taken

$$\lambda = \lim_{N \to \infty} \frac{1}{N} \ln |\frac{\Delta z_1}{\Delta z_0}|. \tag{2.12}$$

This is similar to calculating the eigenvalues of the linearized Poincaré map. In fact, the Lyapunov exponents for higher dimensional systems can also be calculated as the average moduli of the eigenvalues

$$\lambda_i = \lim_{t \to \infty} \frac{1}{T} \ln |m_i(T)|.$$

[2]This only insures that this system does not have an aperiodic behavior but not necessarily sensitivity to initial conditions.

37

where $m_1, m_2 \ldots m_n$ are the eigenvalues of $DP(z)$.

The Lyapunov exponents are closely related to the eigenvalues discussed previously and are calculated by similar means, but there is an important difference. Whereas eigenvalues are usually calculated at a point in state space, such as a fixed point, Lyapunov exponents are usually geometrically averaged along the orbit. The Lyapunov exponents are the average rate of contraction or expansion near the periodic orbit. Knowing how the local Lyapunov exponent varies in space allows one to identify regions of an attractor with good or poor predictability for small initial errors. More about numerical calculations of the Lyapunov exponents can be found in (Parker and Chua, 1989; Sprott, 2003).

2.2.5.3   Spectrum analysis - Fourier analysis

Another important tool in trying to diagnose a bifurcating nonlinear system is the power spectrum analysis. This method studies the frequency content of a solution of a nonlinear ODE. By first taking the time series data and analyzing using fast Fourier transforms (FFT), one can find the dominant frequencies contained in the solution. A period-1 orbit will consist of the fundamental frequency and higher harmonic frequencies in multiples of the fundamental frequency. After a period doubling bifurcation occurs another frequency will join the fundamental one. This frequency will be half the frequency of the fundamental one. After each doubling bifurcation another frequency, and its corresponding harmonics, will join the spectrum.

Part I

Open-Loop Climbing[3]

Chapter 3

DSAC

This chapter explores the DSAC, for Dynamic Single Actuated Climber. We first introduce the system and model assumptions, then derive the equations of motions. The analysis of the mechanism is divided into three sections: local stability, basin of attraction and efficiency calculation. Three interesting results are reported. First the mechanism is extremely robust and stable, even while using open-loop control. This is observed in the local stability of the system and is mostly surprising due to the high robustness of the system manifested in a large basin of attraction. Period doubling bifurcation is observed in simulation and experiments. Another interesting phenomenon that we show is that non-symmetric "limping" climbing is more efficient and more stable in some cases.

3.1 Modeling

3.1.1 System description and modeling assumptions

In simulations and experiments DSAC exhibits stable periodic climbing motions. The goal of our analysis is to produce a model that exhibits behavior similar to that of the experiments and simulations. The DSAC mechanism is planar and consists of two links; the first is the leg which

41
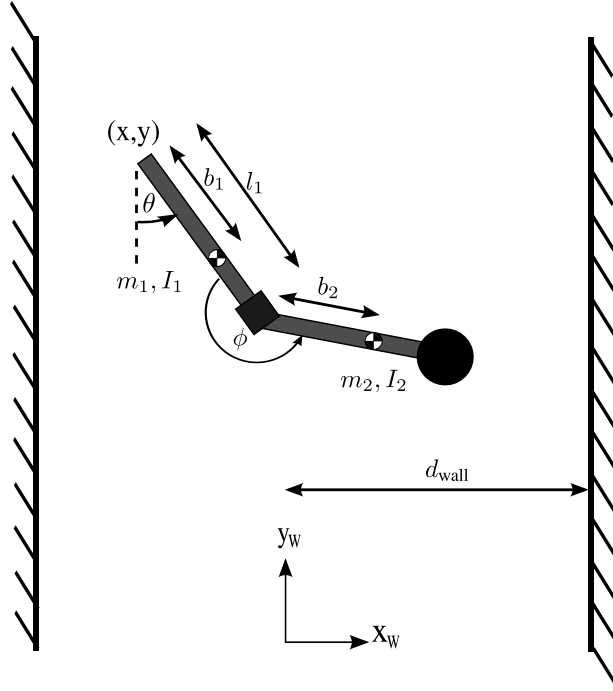
Figure 3.1: Schematics of two link mechanism climbing between two parallel walls.

contacts the wall only at its distal tip. The second link is the main body which is connected to the leg through an actuated revolute joint (Figure 3.1). The leg has mass m_1, moment of inertia I_1, and length l_1. Its CoM is located at a distance b_1 from the contact point with the wall. The body is assumed to have mass m_2, moment of inertia I_2, and CoM located b_2 from the joint connecting it to the leg. The body does not collide with the leg nor with the walls. The cartesian coordinates (x,y) are chosen at the distal end of the leg, the angle of the leg relative to the vertical is θ, and the angle between the two links is ϕ. The motion between the two links is set to be a sinusoid $\phi(t) = A\sin(\omega t)$, where A and ω are the amplitude and angular frequency of the sinusoid, respectively. For simplicity, the inertial frame is centered between the two parallel walls. In fact it is possible to only use five parameters ($I_1 + m_2 l_1^2$, I_2, $m_2 b_2 l_1$, $m_1 b_1 + m_2 l_1$, and $m_2 b_2$) instead of the full set of seven parameters (m_1, m_2, l_1, I_1, I_2, b_1, and b_2) used here (c.f. Dullin (1994)). See Appendix B for full derivation of the equations of motion. To analyze the behavior of the mechanism, the motion is split into three phases: flight, impact, and stance

phase. By using the final state of one phase as the initial values of the next phase we can analyze and simulate the whole climbing motion. Since the environment is symmetric (two parallel walls), we can include a "flip" of coordinates during impact phase, this will enable the equations to always represent a robot leaping from the right towards the left wall.

A few hypotheses and assumptions are used throughout to simplify the analysis. We assume that the impact model is instantaneous and inelastic, where no slipping or rebound occurs. The external forces during the impact can be represented by impulses, which may result in an instantaneous change in the velocities but not in the configuration. Since the actuator has a known sinusoidal trajectory, during the impact we assume the motor can apply an impulsive torque to keep itself on track. Moreover, the angular momentum around the contact point is constant hence the angular velocity of the leg can be calculated.

During the stance phase the leg is in contact with the wall. Due to high friction between the leg and the wall, no sliding will occur and the contact point is treated as a frictionless pin joint. We only consider the gait where the distal end of the leg hits the wall. Although not physical, we assume that during the transition from stance phase back to flight, no slip occurs. From these assumptions we allow the leg angle and angular velocity to be without limits. In the physical mechanism in order to hold these assumptions the leg angle and angular velocity will be in the range of $\theta \in (0 : 1.2\text{rad})$, and $\dot{\theta} \in (-2 : 2\frac{\text{rad}}{\text{s}})$. We derived and simulated the equations of motion using Matlab$^{\text{TM}}$.

3.1.2   General equations of motion

Since the angle between the two links, ϕ, is constrained, it will not be part of the state of the robot. This forced periodic input turns the equations of motion into a nonautonomous system (as discussed in Chapter 2, Eq. 2.9)

$$\dot{z} = f(\hat{z}, t, \mu),\tag{3.1}$$

where the state of the system $\hat{z} = (\hat{q}, \dot{\hat{q}})$ is 6-dimensional, where $\hat{q} = (x, y, \theta) \in \mathbb{R}^2 \times \mathbb{S}^1$ and $\dot{\hat{q}} = (\dot{x}, \dot{y}, \dot{\theta}) \in \mathbb{R}^3$, and $\mu \in \mathbb{R}^n$ is the system's parameters. These parameters include, for example, the link lengths and the distance between walls. Such periodic forced system can then be converted into an autonomous system as shown in Guckenheimer and Holmes (1983) by increasing the dimension by one.

$$\dot{z} = f(z, \mu), \tag{3.2}$$

The state of the system z is now $(q, \dot{q})$, where $q = (x, y, \theta, \tau) \in \mathbb{R}^2 \times \mathbb{S}^1 \times \mathbb{S}^1$, $\tau = \omega t \pmod{2\pi} \in \mathbb{S}^1$, and $\dot{q} = (\dot{x}, \dot{y}, \dot{\theta}) \in \mathbb{R}^3$. The addition of τ comes from the conversion to an autonomous system.

Writing in Lagrange matrix form, as in Eq. 2.5, the equations of motion are

$$M(q)\ddot{q} + h(q, \dot{q}) = \Upsilon - A^T(q)\lambda_{ext}, \tag{3.3}$$

where $M(q) \in \mathbb{R}^{3 \times 3}$ and $h(q, \dot{q}) \in \mathbb{R}^{3 \times 1}$ are the mass matrix and the nonlinear terms matrix, respectively. The vector representing the applied forces and torques is $\Upsilon \in \mathbb{R}^{3 \times 1}$ and λ_{ext} is the contact force with the wall. Since ϕ is a constrained motion there is no need to include input torques ($\Upsilon = 0$). Let $A(q) = \frac{\partial P(q)}{\partial q} \in \mathbb{R}^{2 \times 3}$ be the constraint matrix during contact, where $P(q) = [P_x \quad P_y]^T$ is the point of contact with the walls. Since the point of contact $P(q)$ coincides with our coordinate system, $P(q) = [x \quad y]^T$ and $A(q) = \left(\begin{smallmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{smallmatrix} \right)$. This system is underactuated in the sense that only one actuator exists. This is the motor which connects the leg to the main body. For a detailed derivation of these equations of motion see Appendix B.

As mentioned before, to analyze the behavior of the mechanism, the motion is split into three phases: flight, impact, and stance phase (see Figure 3.2). By using the final state of one phase as the initial values of the next phase we can analyze and simulate the whole climbing motion.

A projection of the phase space portrait, including only θ and $\dot{\theta}$ of a climbing gait from one wall to the other and back to the first, is depicted in Figure 3.3. This motion consists of two consecutive three-phase motions which will now be derived in more detail.

44
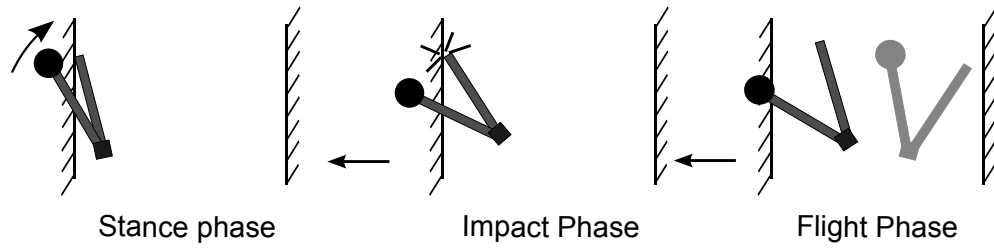
Figure 3.2: Three phases of climbing gait.
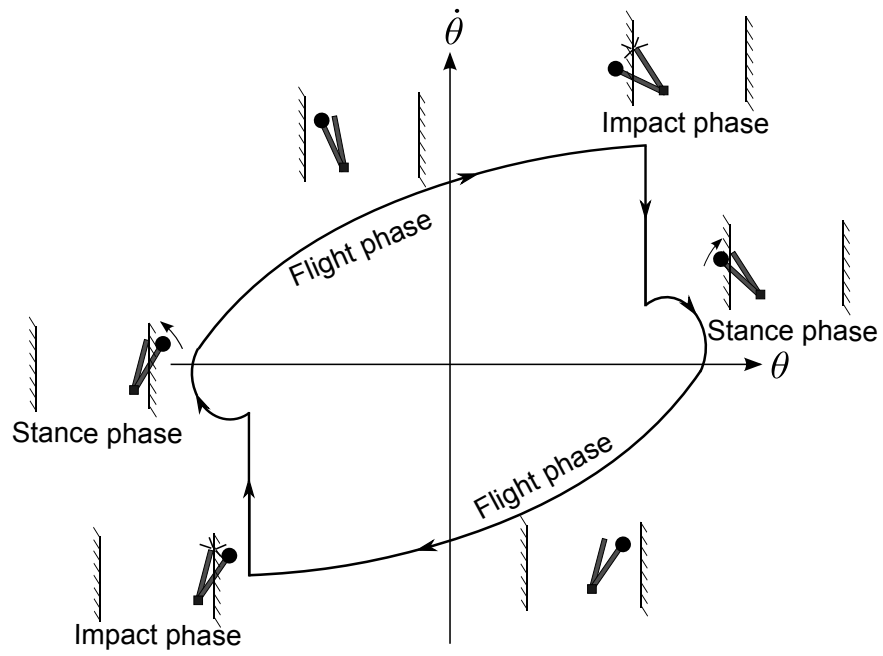
Figure 3.3: Projection of phase plot onto the θ, $\dot{\theta}$ plane of the nominal climbing motion.

3.1.2.1 Free flight phase

During free flight, there are no external forces acting on the mechanism other than gravity. There-fore, $\lambda_{ext} = 0$ and the equations of motion are reduced to:

$$M(q)\ddot{q} + h(q, \dot{q}) = 0, \tag{3.4}$$

3.1.2.2 Impact phase

From the impact assumptions in Section 3.1.1 one can find the equation of conservation of angu-lar momentum around the contact point during the instantaneous time of impact. As described in (Greenwood, 1997), the total angular momentum with respect to point c is

$$H = \sum_{i=1}^{N} I_i \omega_i + \vec{r}_{m_i/c} \times m_i \dot{r}_{m_i}, \tag{3.5}$$

where N is the number of links, $\vec{r}_{m_i/c}$ is the vector from mass i to the point of contact. $\dot{r}_{m_i}$ is the velocity of mass i relative to the inertial frame. I_i is the moment of inertia of link i and ω_i is the angular velocity of link i as depicted in Figure 3.4.

From the previous assumptions we know that during impact, only the velocity and not the configuration changes, hence the only unknown state variable is $\dot{\theta}$. From this equation we can find the new $\dot{\theta}^+$ after the collision. This is done by equating the angular momentum before impact and instantaneously after impact.

$$\dot{\theta}^+ = f(\theta, \phi, \dot{x}^-, \dot{y}^-, \dot{\theta}^-), \tag{3.6}$$

where f maps the angular velocity of the leg from pre-impact to post-impact.

Since the environment is symmetric (two parallel walls), we can include a "flip" of coor-dinates during impact phase, this will enable the equations to always represent a robot leaping off the right towards the left wall. All of the state coordinates other than the y coordinates will change sign. Since, by assumption, no sliding is allowed, the linear velocities will vanish after

46
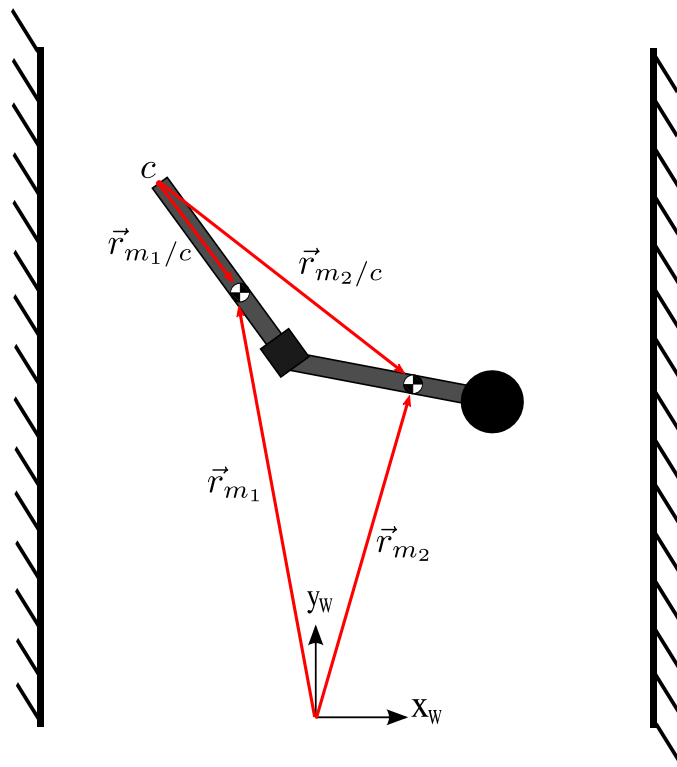
Figure 3.4: Finding angular momentum around contact point c.

impact. Therefore there exists a mapping which describes this whole transition during impact

$$
\begin{bmatrix} x^+ \\ y^+ \\ \theta^+ \\ \dot{x}^+ \\ \dot{y}^+ \\ \dot{\theta}^+ \end{bmatrix} = \begin{bmatrix} -x^- \\ y^- \\ -\theta^- \\ 0 \\ 0 \\ -f(\theta, \phi, \dot{x}^-, \dot{y}^-, \dot{\theta}^-) \end{bmatrix},
\tag{3.7}
$$

where the $^-$ and $^+$ subscripts represent the pre and post impact state variables, respectively.

3.1.2.3 Stance phase

Since we assume that the leg which is in contact with the wall will not slip, the external (contact) force can be added and the system can be described as

$$
M(q)\ddot{q} + h(q, \dot{q}) - A^T(q)\lambda_{ext} = 0,
\tag{3.8}
$$

where $\lambda_{ext} = [\lambda_n, \lambda_t]^T$ are the normal and tangential contact forces between the tip of the leg and the wall. As described in Eq. 3.3, the constraint matrix $A(q)$ is $A(q) = \left(\begin{smallmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{smallmatrix} \right)$. We add an extra set of equations to find the time when the normal contact force changes sign. This is the instant when the leg loses contact with the wall and the mechanism transitions into flight phase. As shown in section 2.1.2, we can apply some simple manipulation to add the Lagrange multiplier constraint and find the external force equation as a function of the state.

$$
\lambda_{ext}(q, \dot{q}) = \left(A(q)M(q)^{-1}A(q)^T \right)^{-1} \left(\dot{A}(q)\dot{q} - A(q)M(q)^{-1}h(q, \dot{q}) \right).
\tag{3.9}
$$

When λ_n changes sign, the stance phase terminates and the flight phase begins. Using the final conditions of the stance phase as the initial conditions of the flight phase, we can continue and simulate the next three phases. The three phases including the flip during impact phase are shown in Figure 3.5.

48

We use the method of nondimensionalizing equations of motion to find the nondimensional parameters of our equations of motion. This method is not well known nor often used in the robotics community but can be beneficial. The motivation and procedure is given in Appendix A. The derivation of the DSAC nondimensional equations of motion are presented in Appendix B. The characteristic length and time for this non-unique set we chosen to be d_{wall} and $\frac{1}{\omega}$, respectively which results in the set given below. The nondimensionalization reduces the number of parameters of the system from eleven to eight.

$$\mu = \frac{m_2}{m_1}, \beta = \frac{b_1}{l_1}, \gamma = \frac{b_2}{b_1}, \delta = \frac{l_1}{d_{\text{wall}}}, \Omega = \frac{g}{\omega^2 d_{\text{wall}}}, A,$$
$$\rho_1 = \frac{I_1}{d_{\text{wall}}^2(m_1 + m_2)}, \quad \rho_2 = \frac{I_2}{d_{\text{wall}}^2(m_1 + m_2)}.$$

(3.10)

These nondimensional parameters provide some valuable information without even observing or solving the equations of motion.

3.2 Approach to Analysis

3.2.1 A typical climbing motion

Figure 3.3 depicted a full climbing cycle between one wall to the other and back to the first. Figure 3.6 depicts a typical period-1 phase plot cycle for the new climbing motion after adding the "flip", as in Eq. 3.7. This plot is a phase plot of θ and $\dot{\theta}$. Although this is not the full state

Post-Impact   Pre-Impact

Stance phase   Impact Phase   Flight Phase
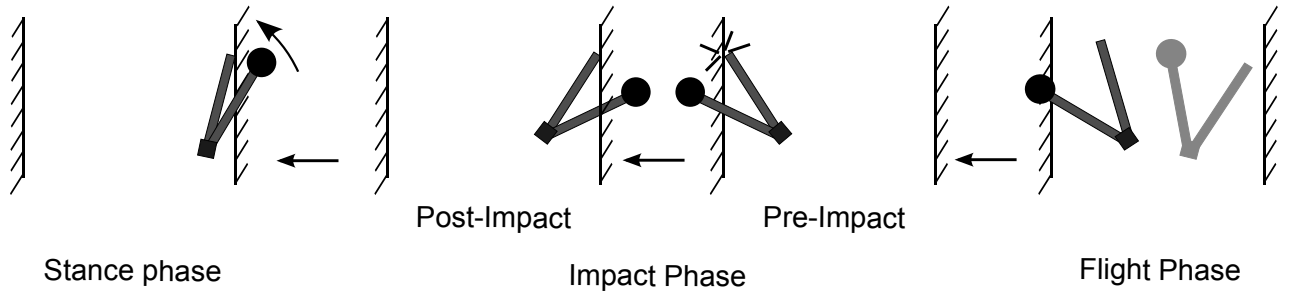
Figure 3.5: Three phases including flip during impact phase.

49

space but only a projection on the $\theta, \dot{\theta}$ plane, this phase plot portrays the important information of the climbing cycle. In fact, we will later show that these two state variables together with a variable corresponding to phase are all the information needed to portray the motion of the mechanism.
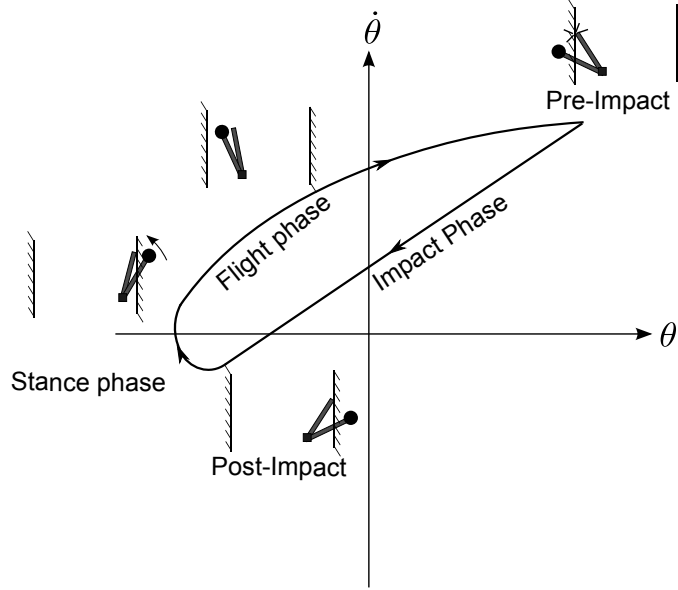
Figure 3.6: phase plot of a typical climbing motion with the flipping action of impact phase.

3.2.2 Open-loop stability

This section explains the stability investigation of the climbing mechanism. There are several approaches to investigate nonlinear systems. One, which for example was taken by McGeer (1990b), is to linearize the governing equations of motion about an equilibrium state. This might allow us to explicitly integrate the equations of motion. There are two problems with this approach. The first is that the solutions are only valid in a small region around the linearized state. This accounts for the loss of important information and for inaccurate stability models (as shown in Goswami et al., 1998). The second problem is that in our mechanism, in order to calculate the time of flight, a transcendental equation must be solved. This forces us to solve the equation numerically. The

second approach, which is used here, is to preserve the full, nonlinear hybrid equations of the system. The main disadvantage in this approach is that we must rely on numerical solutions.

Using the definitions of orbital stability from Section 2.2.1, we will explore the regions where the mechanism is stable. One should note that we are only interested in the orbital stability, i.e., the stability related to a closed orbit (or limit cycle). Similar to other locomoting systems, one state variable is related to the progression of the mechanism. In walking machines this is the horizontal coordinate, whereas in our system it is the vertical displacement (y). Since our system locomotes, this variable is not cyclic hence we are not interested in finding its orbital stability. One might either ignore this variable, as done in most related work, or show that the equations of motion are invariant to this variable, therefore will not be cyclic in general. The next theorem will explain why invariance of the displacement variable corresponds to a non-cyclic motion.

Theorem 3.1. *Consider an n dimensional system of ODE with state $\{z_1, z_2, \ldots, z_{2n}\}$. If the system is invariant to one of the generalized coordinates, i.e., $\dot{z} = f(z_2, z_3, \ldots, z_{2n})$, then*

1. *The system can be solved by first solving the reduced $n-1$ dimensional system $\dot{\zeta} = f(\zeta)$, where $\zeta = (z_2, z_3, \ldots, z_{2n})$, followed by solving the single ODE $\dot{z}_1 = f(\zeta)$.*

2. *The invariant variable z_1 can never be a non zero-mean periodic on average, i.e., on average the variable will either increase or decrease.*

Proof. The sketch of the proof is straightforward. Since the system is invariant to z_1, it is possible to decouple the system into a reduced $n-1$ second order differential equations together with a single first order differential equations by substituting $u = \dot{z}_1$. This process is similar to reduction of order of a differential equation when a dependent variable is missing (e.g., Boyce and DiPrima, 2001). Second, after solving the reduced system, numerically in our case, one can simply integrate u to find z_1. The consequence of integrating $\dot{z}_1$ is that unless $\dot{z}_1$ has a zero mean z_1 will always either increase or decrease, hence will not be cyclic. □

Theorem 3.1 is significant to our system because our equations of motion are not dependent on the vertical displacement (y). Physically, since the walls are vertical, placing the robot in a

different y location should not change the dynamics of the system other than initial condition. Moreover, since the robot climbs it does not make sense to find the orbital stability in that direction because it is not periodic. Therefore, the entire state space can be decomposed into two spaces, the y climbing and the rest of the state which can be periodic and stable. For stability analysis we can use Theorem 3.1 and exclude the y variable. We will only use y when we are interested in finding how much the robot climbed.

3.2.3   Poincaré map and corresponding Poincaré section

Using Definition 2.3, we can define the Poincaré map from the Poincaré section mapped back to this section by $\mathbf{P}$

$$z_{k+1} = \mathbf{P}(z_k), \tag{3.11}$$

where $\mathbf{P}$ is the map, z_k and z_{k+1} are states in the reduced spaces on the Poincaré section before and after the map, respectively. For this system, a convenient Poincaré section is the instant of release from the wall, i.e., the transition from stance to flight phase. This occurs when the normal contact force λ_n passes through zero from negative to positive. Because during stance phase the distal part of the leg is touching the wall ($x = d_{\text{wall}}$), no rebound ($\dot{x} = 0$) or slippage ($\dot{y} = 0$) occurs , we can define a reduced dimensional hyperplane Σ as the Poincaré section

$$
\begin{aligned}
\Sigma = & \{(x, \theta, \dot{x}, \dot{y}, \dot{\theta}, \tau) \in \mathbb{R}^4 \times \mathbb{S}^1 \times \mathbb{S}^1 \\
& \mid x = d_{\text{wall}}, \dot{x} = 0, \dot{y} = 0, \lambda_n(z, \tau) = 0\}
\end{aligned}
\tag{3.12}
$$

This Poincaré section reduces the dimension of the Poincaré map to three, including only $\theta, \dot{\theta}, \tau$.

$$\mathbf{P} : (\theta, \dot{\theta}, \tau) \in \mathbb{S}^1 \times \mathbb{R}^2 \to (\theta, \dot{\theta}, \tau) \in \mathbb{S}^1 \times \mathbb{R}^2,$$

If the mechanism reaches the wall during the climbing cycle, then the state z must lie on Σ. Other motions that do not reach the wall cannot be analyzed using this method, however, they are not of interest since pushing off the wall is needed for stable climbing. We will also note that y, the

52

vertical climbing direction, is not included in the definition of the Poincaré map or the Poincaré section since it is not part of the limit cycle, and we do not want to stabilize it.

Although the Poincaré section reduces the state tremendously (from eight to three), it is not trivial to calculate the exact transition since the contact forces need to be calculated. In this thesis we simplify the section even further by assuming that the transition between stance and flight phases occurs when the acceleration of the swinging leg (ϕ) changes sign, i.e., when $\ddot{\phi} = -A\omega^2 \sin(\omega t) = 0$. This event occurs when $\tau = \omega t \pmod{2\pi} = 2\pi$. The new Poincaré section can therefore be defined as

$$\begin{aligned} \Sigma = &\{(x, \theta, \dot{x}, \dot{y}, \dot{\theta}, \tau) \in \mathbb{R}^4 \times \mathbb{S}^1 \times \mathbb{S}^1 \\ &\mid x = d_{\text{wall}}, \dot{x} = 0, \dot{y} = 0, \tau = 2\pi\} \end{aligned} \tag{3.13}$$

In this Poincaré section all state variables are constrained, except θ and $\dot{\theta}$. Therefore, the Poincaré map is defined as

$$\mathbf{P} : (\theta, \dot{\theta}) \in \mathbb{S}^1 \times \mathbb{R} \rightarrow (\theta, \dot{\theta}) \in \mathbb{S}^1 \times \mathbb{R},$$

including only $\theta, \dot{\theta}$. We have compared both Poincaré sections (Eq. 3.12 and Eq. 3.13) and decided to use the lower dimensional one since it is almost identical and is much easier to analyze and to graphically present.

3.2.4 Local stability

We refer to stability of the climbing mechanism as the local orbital stability, i.e., the stability of an orbit in phase space around a fixed point on the Poincaré section. In order to find this kind of stability we must first find the fixed point of the Poincaré map, then linearize the Poincaré map around the fixed point, and finally find the characteristic multipliers which are the eigenvalues of this linearized Poincaré map (Jacobian). These characteristic multipliers correspond to the rate of convergence (or divergence) from the limit cycle. For an orbitally stable cycle, the characteristic multipliers must lie within the unit circle on the complex plane; i.e., their moduli are strictly

less than one. This investigation is conducted numerically by first using the Newton-Raphson method to find the fixed point, and then calculating the Jacobian (linearized Poincaré map) and its eigenvalues numerically.

3.2.4.1 Fixed point search

The fixed point is the initial state of the mechanism that will map back to itself after one Poincaré map. Thus, we need to solve the equation

$$\mathbf{F}(z) \triangleq z - \mathbf{P}(z) = 0. \tag{3.14}$$

This search is done by fixing the mechanism parameters and using the multidimensional Newton-Raphson method. The solution is not guaranteed and may not be unique. Note that during the Newton-Raphson search we need to solve the Poincaré map, i.e., forward simulate the three phases. During the flight phase, if the mechanism does not reach the wall after a certain integration time it is concluded that there is no fixed point. In fact, even if it were a fixed point, it will not be of interest for our climbing analysis because it will likely not be climbing at all.

3.2.4.2 Linearized Poincaré map and eigenvalues

The linearized Poincaré map around the fixed point which was previously found, is the Jacobian of the map.

$$\nabla \mathbf{P} = \left[\frac{\partial \mathbf{P}}{\partial \theta} \quad \frac{\partial \mathbf{P}}{\partial \dot{\theta}} \right]^{T}. \tag{3.15}$$

Calculating the elements of the Jacobian is done numerically using either the central difference or the forward difference derivative approximation. The central difference can be slightly more accurate but requires more evaluations of the Poincaré map $\mathbf{P}$. Therefore, the simplified forward difference was chosen, which finds the elements of the Jacobian by perturbing the state by a small scalar dz in direction i, mapping $\mathbf{P}(z_1, \ldots, z_i + dz, \ldots, z_n)$, finding the difference between it and the unperturbed map, and finally taking the ratio to the perturbed amount. The ith element will

therefore be

$$\frac{\partial \mathbf{P}}{\partial z_i} = \frac{\mathbf{P}(z_1, \ldots, z_i + dz, \ldots, z_n) - \mathbf{P}(z)}{dz} \tag{3.16}$$

for our system the Jacobian will be

$$\nabla \mathbf{P} = \left[\frac{\mathbf{P}(\theta + dz, \dot{\theta}) - \mathbf{P}(\theta, \dot{\theta})}{dz} \quad \frac{\mathbf{P}(\theta, \dot{\theta} + dz) - \mathbf{P}(\theta, \dot{\theta})}{dz} \right]^T \tag{3.17}$$

The mapping $\mathbf{P}$ is locally orbitally stable if the Poincaré map of a perturbed state is closer to the fixed point than the perturbed state. This property can be viewed as the contraction of the phase space around the limit cycle. This means that the magnitude of the characteristic multipliers (eigenvalues) of $\mathbf{P}$ at the fixed point are strictly less than one, as discussed in Chapter 2. The eigenvalue calculations were done numerically using Matlab$^{\text{TM}}$. To analyze and categorize the bifurcations of the system, we will plot the characteristic multipliers locus and check for bifurcation while changing a parameter continuously.

3.2.5 Efficiency

In order to find the efficiency of a specific climbing gait, we calculate the total work done by the motor. To do so, we first need to find the required torque that the motor has to exert in order to keep the desired sinusoidal. To do so we look at the free body diagram of the main body (Figure 3.7). The next equations which states the change of angular momentum is equal to the sum of torques at the hinge enables us to extract the torque applied by the motor.

$$\dot{\vec{H}}_{/O} = \Sigma \vec{M}_{/O} \tag{3.18}$$

where O is the location of the hinge, $\Sigma \vec{M}_{/O}$ is the sum of torques applied on the body, and $\dot{\vec{H}}_{/O}$ is the rate of change of the angular momentum about the hinge. More explicitly, the rate of change of the angular momentum and the sum of torques are:

$$\dot{\vec{H}}_{/O} = m_2 \vec{r}_{G_2/O} \times \vec{a}_{G_2/W} + \ddot{\theta}_2 I_2 \hat{k} \tag{3.19}$$

55
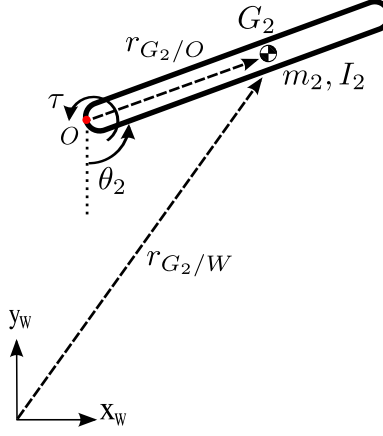
Figure 3.7: Free body diagram of main body.

$$\Sigma \vec{M}_{/O} = r_{G_2/O} \times (-m_2 g)\hat{j} + \tau \hat{k}, \qquad (3.20)$$

where $r_{G_2/O}$ is the vector from the hinge to the CoM of the main body, $a_{G_2/W}$ is the acceleration of the main body relative to the inertial frame, and $\hat{j}, \hat{k}$ are the y and z directions, respectively. Inserting Eqs. 4.2,4.3 into Eq. 3.18 we can find τ:

$$\tau = m_2 \vec{r}_{G_2/O} \times \vec{a}_{G_2/W} + \ddot{\theta}_2 I_2 \hat{k} - r_{G_2/O} \times (-m_2 g)\hat{j}. \qquad (3.21)$$

This process of finding the torque is in essence the inverse dynamics solution of the system. By first constraining the system to move in a sinusoid, we solve the equations of motion to find the state of the bodies and finally find the torque required to achieve these accelerations. The instantaneous power is $\tau \cdot \dot{\phi}$, where $\dot{\phi}$ is the relative angular velocity of the two links, i.e., in our case $\dot{\phi} = A\,\omega\,\cos(\omega t)$. Finally, the total work is

$$W = \int_0^T \tau \cdot \dot{\phi}\, dt = \int_0^T \tau\, A\,\omega\,\cos(\omega t)\, dt \qquad (3.22)$$

The efficiency of the system, η, is calculated as $\eta = \frac{\Delta E}{W}$, where ΔE is the change of energy during one Poincaré map, and W is the work calculated using Eq. 3.22.

3.2.6 Basin-of-attraction

Section 3.2.4 discusses the stability analysis of the climbing mechanism. Specifically, the local orbital stability was investigated by first finding the fixed point, then linearizing around this point and lastly looking for the eigenvalues which correspond to the rate of convergence (or divergence) from the limit cycle. This stability criterion is only applicable locally around the fixed point. This section will approximate the basin-of-attraction of the attractors. That is, what set of initial conditions will converge to one of the fixed points.

This investigation can be done by discretizing the state space and forward simulating these initial conditions until they converge on an attractor. This method is timely and computationally consuming. Alternatively, a different technique called cell mapping (or cell to cell mapping) can give a relatively good estimate of regions of attractions in the state space. The cell mapping technique, described in full in Hsu (1997), assumes that the topological structure of the flow changes continuously, hence neighboring points behave in a similar manner. Using this assumption, the phase space is discretized into a large number of small cells and the entire cell is represented by its center. One Poincaré map is numerically calculated once for the center of these cells and the information is recorded. Cells mapped outside of the discretized area are marked as mapped to a "sink cell". Sink cells are also mapped back to themselves. All the dynamic information, up to the precision of the grid division, is now contained in these simple pointers. It is now possible to iterate these pointers to find periodic cycles.

Instead of calculating the complicated Poincaré map from each initial condition until convergence, which can be about 10,000 cycles, this method only computes the Poincaré map once for each cell. The main disadvantage is that depending on the resolution of the discretization, this method can falsely classify periodicity, though, it usually provides a good idea of where the relevant regions of attractions are. We have implemented a few modifications for this methods, including changing the number of Poincaré maps that are calculated. When approximately 10 Poincaré maps are initially calculated for each of the cells, the resulting basin of attraction is

much more accurate and clears most, if not all, of the falsely classified attractors. Of course this modification will not be able to classify the difference between different lower order periods if two stable attractors coexist.

3.3 Analysis Results

This section will present a few interesting phenomena which occur in specific mechanism parameters. We first show in simulation that stable, open-loop climbing motion do exist. Moreover, these motions have a relatively large basin of attraction. Period doubling occurs when controls or mechanism's parameters are varied, for example, while varying the sinusoid frequency (ω) or leg lengths ratio (γ). Interestingly, in some cases the non-symmetric period-2 are more stable, more efficient and even better climbers than the symmetric period-1 motions.

The results shown here are for the mechanism and environment parameters given in Table 3.1. Notice that the effective gravity is a tenth of the normal gravity. This will later help us in the experimental section to obtain interesting climbing phenomena using slower motor speeds.

3.3.1 Local orbital stability - bifurcations

This section will use the process described above to first explore the orbital stability characteristics of a typical DSAC mechanism. In order to more easily and quickly find the mechanism parameters and control inputs where the interesting bifurcations occur, we plot the characteristic multipliers locus while varying one of the parameters. As discussed in Section 2.2.5 and 3.2.4, the different locations where the eigenvalues (characteristic multipliers) of the linearized Poincaré map cross the unit circle implies different bifurcations. One major advantage of using this characteristic multipliers locus method over forward simulating and plotting the bifurcation diagram is that with the latter method it is important to keep the step size of the varied parameter small while in the former it is not as sensitive to step size. Also, with the characteristic multipliers locus it is easier to distinguish between different types of bifurcations.

3.3.1.1 Varying angular frequency inputs (ω) for different leg inertia (I_1)

Figure 3.8 and Figure 3.9 depicts the characteristic multipliers locus and bifurcation plots while varying ω (angular frequency input) for three different leg inertia (I_1). As can be seen for rela-

Table 3.1: DSAC parameters for results section.

Dimensional Parameters		
Parameter	Description	Value
m_1	leg mass	0.3 kg
m_2	body mass	0.7 kg
l_1	leg length	0.075 m
b_1	CoM of leg	0.075 m
b_2	CoM of body	0.06 m[†]
I_1	leg inertia	$1 \cdot 10^{-5}$ kg m^2[†]
I_2	body inertia	0 kg m^2
d_{wall}	half wall width	0.045 m
g	gravitational acceleration	0.9807 $\frac{\text{m}}{\text{s}^2}$
ω	sinusoid frequency	15 $\frac{\text{rad}}{\text{s}}$[†]
Nondimensional Parameters		
μ	Mass ratio $\frac{m_2}{m_1}$	$2\frac{1}{3}$
β	CoM location of leg $\frac{b_1}{l_1}$	1
γ	Link length ratio $\frac{b_2}{b_1}$	0.8[†]
δ	Leg to wall gap ratio $\frac{l_1}{d_{\text{wall}}}$	$1\frac{2}{3}$
ρ_1	Nondimensional inertia $\frac{I_1}{d_{\text{wall}}^2(m_1+m_2)}$[†]	0.0049
ρ_2	Nondimensional inertia $\frac{I_2}{d_{\text{wall}}^2(m_1+m_2)}$	0
Ω	Nondimensional gravity $\frac{g}{\omega^2 d_{\text{wall}}}$	3.2131[†]
A	Amplitude	0.8[†]
Parameters marked with [†] are varied in the current analysis		

60

tively high leg inertia $I_1 = 1 \cdot 10^{-3}$ (Figure 3.8(a)), the period-1 characteristic multiplier exits the unit circle at -1 and a period-2 (red dot) appears. This indicates a period doubling bifurcation. As we continue to vary ω the period-2 characteristic multiplier exit the unit circle as complex conjugate, suggesting that a secondary Hopf bifurcation which results in a quasiperiodic solution. As we even further vary ω a period-10 appears. Figure 3.8(b) depicts a characteristic multipliers locus plot for a smaller leg inertia, $I_1 = 3.3 \cdot 10^{-4}$, where another period doubling occurs from period-2 to period-4. Once again a secondary Hopf bifurcation occurs, this time after the period-4. Figure 3.8(c) depicts a characteristic multipliers locus plot for an even smaller leg inertia, $I_1 = 2 \cdot 10^{-4}$, where a third period doubling occurs from period-4 to period-8. The period doubling route continues until the numerical accuracy is not adequate to further distinguish these bifurcations.

3.3.1.2   Varying CoM location (γ) for different leg inertia (I_1)

Figure 3.10 and Figure 3.11 depicts the characteristic multipliers locus and bifurcation plots while varying γ (CoM location) for three different leg inertia (I_1). As can be seen for relatively high leg inertia $I_1 = 1 \cdot 10^{-3}$ (Figure 3.10(a)), the period-1 characteristic multiplier exits the unit circle at -1 and a period-2 (red dot) appears. This indicates a period doubling bifurcation. The stable period-1 characteristic multiplier then reenters the unit circle. Figure 3.10(b) depicts a characteristic multipliers locus plot for $I_1 = 3 \cdot 10^{-4}$, where another period doubling occurs from period-2 to period-4, however before the period-4 continues to bifurcate the period-2 and then period-1 cycles becomes stable again. Figure 3.10(c) depicts a characteristic multipliers locus plot for an even smaller leg inertia, $I_1 = 1 \cdot 10^{-4}$, where a third period doubling occurs from period-4 to period-8. Similar to the graphs shown earlier for change in ω (Figure 3.9), the period doubling route continues to what seems like a chaotic-like solution. We will further investigate this period doubling later in this chapter. As can be seen, the larger the leg inertia is, the less the system is prune to period doubling and will stay at lower periods.

(a) $I_1 = 1 \cdot 10^{-3}$ (b) $I_1 = 3.3 \cdot 10^{-4}$ (c) $I_1 = 2 \cdot 10^{-4}$
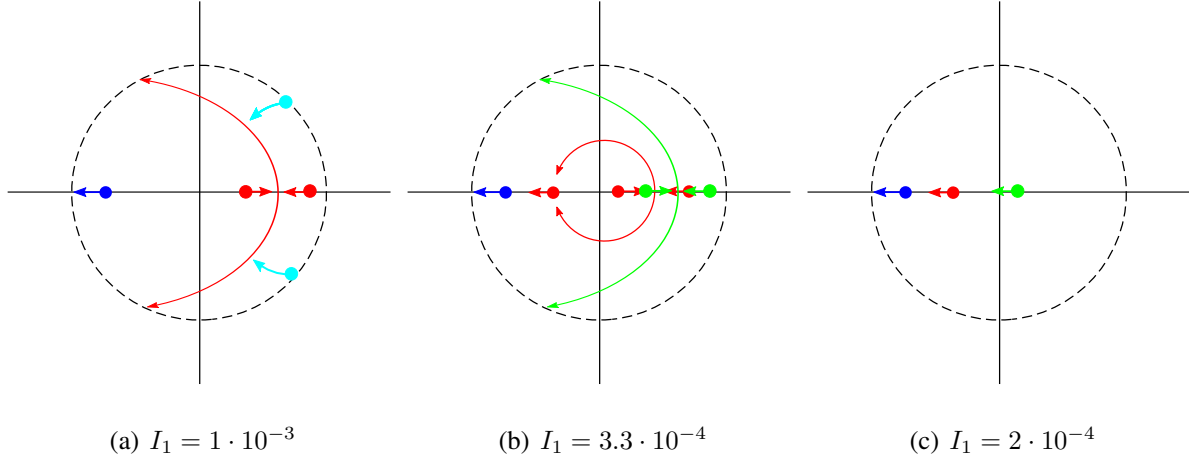
Figure 3.8: characteristic multipliers locus plots while varying ω for three different leg inertia (I_1). Different colors represent different solution. Blue - period-1, Red - period-2, Green - period-4 and Cyan - period-10. As can be seen for relatively high leg inertia $I_1 = 1 \cdot 10^{-3}$ (3.8(a)), the period-1 characteristic multiplier (blue dot) exit from -1 and a period-2 characteristic multipliers (red dots) form. The period-2 characteristic multiplier then exit the unit circle as complex conjugate representing a quasiperiodic solution. Finally a period-10 characteristic multiplier enters the unit circle. Figure 3.8(b) depicts a characteristic multipliers locus plot for a lower leg inertia ($I_1 = 3.3 \cdot 10^{-4}$) where another period doubling occurs from period-2 to period-4, however before the period-4 continues to bifurcate the period-4 solution bifurcates into a quasiperiodic solution. Figure 3.8(c) depicts a characteristic multipliers locus plot for an even smaller leg inertia ($I_1 = 2 \cdot 10^{-4}$) where the period doubling route continues.

(a) $I_1 = 1 \cdot 10^{-3}$ (b) $I_1 = 3.3 \cdot 10^{-4}$ (c) $I_1 = 2 \cdot 10^{-4}$
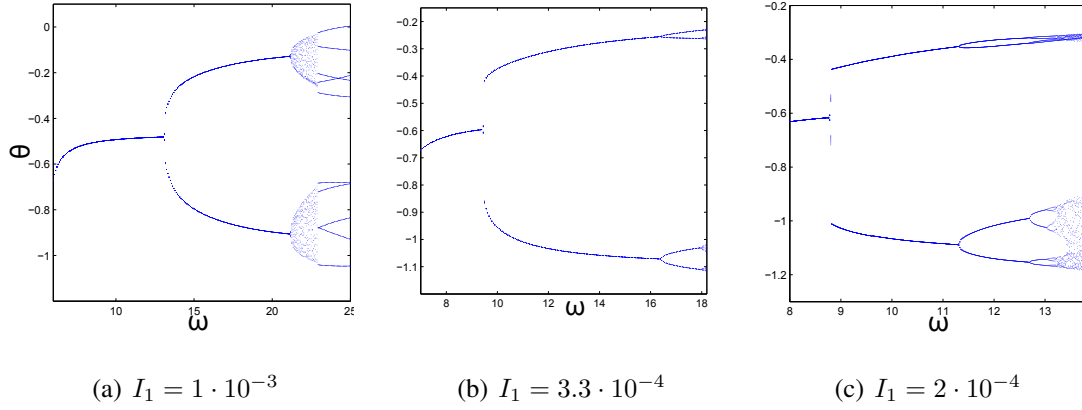
Figure 3.9: Bifurcation plots while varying ω for three different leg inertia (I_1). Corresponding to Figure 3.8, for relatively high leg inertia $I_1 = 1 \cdot 10^{-3}$ (3.9(a)), the period-1 bifurcates into a period-2 then into a quasiperiodic solution and finally into a period-10 solution. Figure 3.8(b) depicts the bifurcation plot for $I_1 = 3.3 \cdot 10^{-4}$, where another period doubling occurs from period-2 to period-4 and then to the quasiperiodic solution. Figure 3.8(c) depicts a bifurcation plot for an even smaller leg inertia ($I_1 = 2 \cdot 10^{-4}$) where a third period doubling occurs from period-4 to period-8 and continues into a period doubling bifurcation cascade.

(a) $I_1 = 1 \cdot 10^{-3}$ (b) $I_1 = 3 \cdot 10^{-4}$ (c) $I_1 = 1 \cdot 10^{-4}$
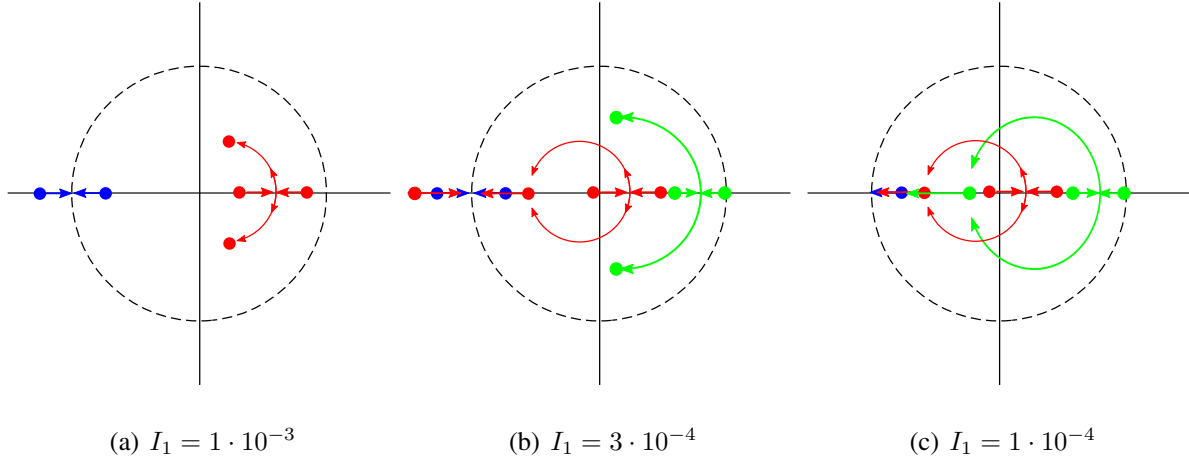
Figure 3.10: characteristic multipliers locus plots while varying γ (CoM location) for three different leg inertia (I_1). Different colors represent different solution. Blue - period-1, Red - period-2, and Green - period-4. As can be seen for relatively high leg inertia $I_1 = 1 \cdot 10^{-3}$ (a) the period-1 characteristic multiplier exits the unit circle at -1 and a period-2 (red dot) appears. This indicates a period doubling bifurcation. (b) Depicts a characteristic multipliers locus plot for $I_1 = 3 \cdot 10^{-4}$, where another period doubling occurs from period-2 to period-4, however before the period-4 continues to bifurcate the period-2 cycle becomes stable again. (c) Depicts a characteristic multipliers locus plot for an even smaller leg inertia, $I_1 = 1 \cdot 10^{-4}$, where a third period doubling occurs from period-4 to period-8.

(a) $I_1 = 1 \cdot 10^{-3}$ (b) $I_1 = 3 \cdot 10^{-4}$ (c) $I_1 = 1 \cdot 10^{-4}$
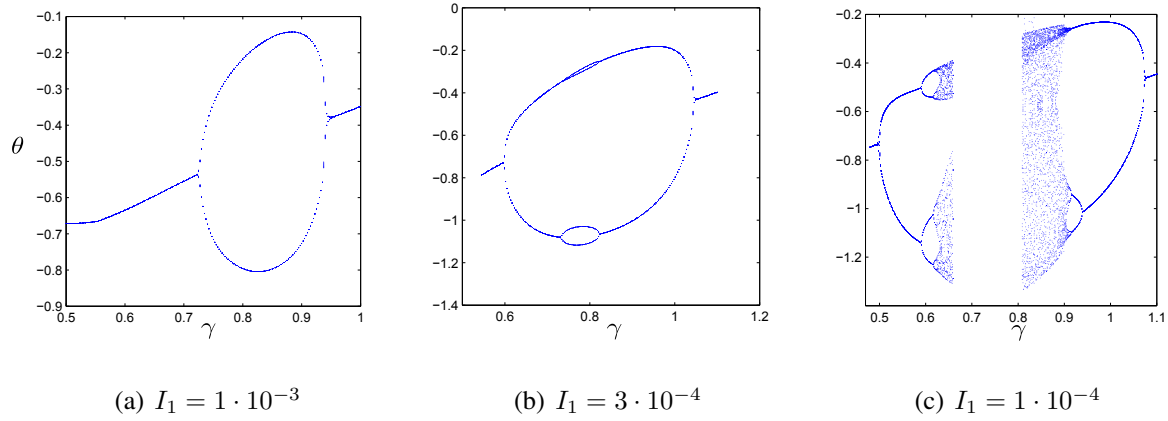
Figure 3.11: Bifurcation plots while varying γ (CoM location) for three different leg inertia (I_1). Corresponding to Figure 3.10, for relatively high leg inertia $I_1 = 1 \cdot 10^{-3}$ (a), the period-1 bifurcates into a period-2 and then back into a period-1. (b) Depicts the bifurcation plot for $I_1 = 3 \cdot 10^{-4}$, where another period doubling occurs from period-2 to period-4, however before the period-4 continues to bifurcate the period-2 cycle becomes stable again. (c) Depicts a bifurcation plot for an even smaller leg inertia, $I_1 = 1 \cdot 10^{-4}$, where a third period doubling occurs from period-4 to period-8 and continues into a period doubling bifurcation cascade. No solution was found in the blank area in the middle.

3.3.1.3 Power spectrum analysis

We use the power spectrum analysis tool to further analyze the bifurcation plots from Figure 3.9 and Figure 3.11. The power spectrum figures were plotted using Welch's power spectral density method and a Hamming window (see Matlab$^{\text{TM}}$). Figure 3.12 samples four input frequencies from Figure 3.9(a) and plots the phase plot together with the Poincaré point marked with red 'x' and the power spectral density (PSD) plot. In Figure 3.12(first and second rows) the period doubling from period-1 ($\omega = 10\frac{\text{rad}}{\text{s}}$) to period-2 ($\omega = 15\frac{\text{rad}}{\text{s}}$) is evident by noticing that an additional frequency with half of the fundamental one was added to the spectrum on the second row. Notice that multiples of the fundamental frequency exists as harmonics. In Figure 3.12(third and fourth rows), we can see two interesting phenomena that can be further explained in the closeup figures of Figure 3.12(c). In the left closeup corresponding to the region close to $\omega = 22\frac{\text{rad}}{\text{s}}$, we can see that the points on the Poincaré surface trace a curve, corresponding to quasiperiodic motion. In the right closeup, corresponding to the region close to $\omega = 25\frac{\text{rad}}{\text{s}}$, one of the two regions of five points on the Poincaré section are shown, corresponding to a period-10 motion.

Similar to the previous power spectral analysis, Figure 3.13 is used to further investigate the bifurcations of Figure 3.11(c). This plot depicts the phase plot together with the Poincaré point marked with red 'x' points on the Poincaré section (3.13(a)) and the PSD plots (3.13(b)) for different leg length ratio γ (with largest γ on top). For the period-1 motion (top of plot, γ=1.1) only the fundamental frequency (and its harmonics) appears on the PSD. Note that the fundamental frequency is normalized. On the corresponding Poincaré section only one point appears. On the second row (γ=1), an additional frequency appears. This frequency which is half of the fundamental one, corresponds to the first period doubling bifurcation. Once again, on the Poincaré section, two points appear. The bifurcations continue with period-4 (γ=0.93) and period-8 (γ=0.915) on the next rows. The bottom row (γ=0.89) begins to show evidence of the chaotic region. Figure 3.13(c) shows closeups of the Poincaré plots for γ=0.89 revealing stretching and folding structure characteristic to chaotic-like strange attractor.
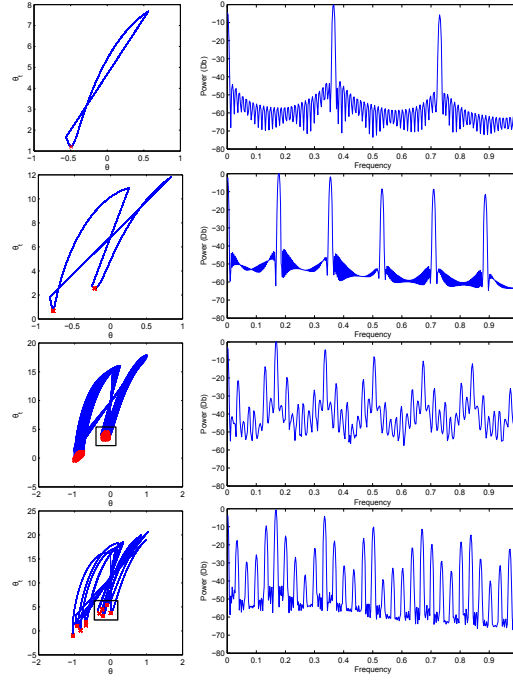
(a) Phase plot   (b) Power Spectrum
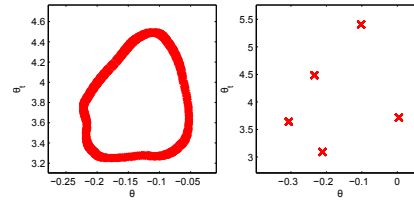
(c) Poincaré close up

Figure 3.12: Varying ω. (a) Phase plot (Poincaré points marked with red 'x'), (b) Power spectrum. Each row represents a different input frequency: $\omega = 10\frac{\text{rad}}{\text{s}}$, $\omega = 15\frac{\text{rad}}{\text{s}}$, $\omega = 22\frac{\text{rad}}{\text{s}}$, $\omega = 25\frac{\text{rad}}{\text{s}}$. The change in number of points on the Poincaré plots (marked with x) together with the (nondimensional) frequency in the power spectrum which is half of the fundamental one, reveal a period doubling bifurcation from $\omega = 10\frac{\text{rad}}{\text{s}}$ to $\omega = 15\frac{\text{rad}}{\text{s}}$. $\omega = 22\frac{\text{rad}}{\text{s}}$ and $\omega = 25\frac{\text{rad}}{\text{s}}$ reveal a quasi-periodic verified by the closed curve on the Poincaré section in the closeup in (c)(left). The closeup of Poincaré plot of $\omega = 25\frac{\text{rad}}{\text{s}}$ reveals a period-10 solution (the closeups are on one of the two clusters).

67

(a) phase plots   (b) Power spectrum

(c) Poincaré plots closeup
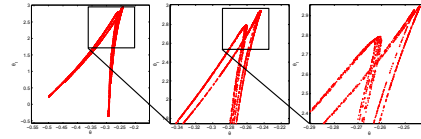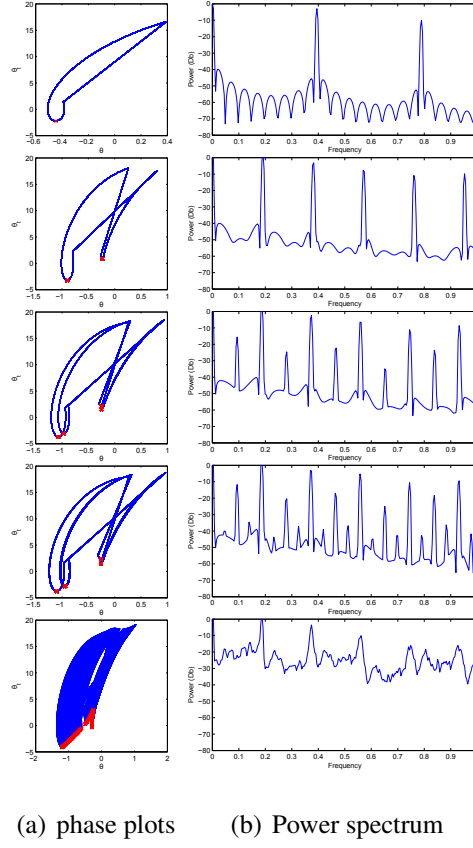
Figure 3.13: Varying γ. (a) Phase plot (Poincaré points marked with red 'x'), (b) Power spectrum. Each row represents a different CoM location ratio (γ). From top to bottom: γ=1.1, γ=1, γ=0.93, γ=0.915, γ=0.89. The Poincaré plots and the PSD reveal period doubling bifurcation. (c) Closeups of Poincaré plots for γ=0.89 revealing stretching and folding structure characteristic to chaotic-like strange attractor.

3.3.1.4 Lyapunov exponents

As mentioned in Chapter 2, a useful tool for analyzing nonlinear systems, specifically bifurcations and chaotic regions, is the Lyapunov exponents. Lyapunov exponents give the average sensitivity of initial conditions along an orbit. The region around $\omega = 23\frac{\text{rad}}{\text{s}}$ in the bifurcation plot in Figure 3.8(a) is suspicious as either being quasiperiodic or chaotic. Using the Poincaré map and spectral density in Figures 3.12, this region is identified as being a quasiperiodic and not chaotic. To verify this, we plot the largest Lyapunov exponent and examine whether there is a high sensitivity to initial condition in this region, corresponding to chaos. A plot of the largest Lyapunov exponent of the same parameters as depicted in Figure 3.8(a) is shown in Figure 3.14. We can then assume that this suspicious region is not chaotic but quasiperiodic as was assumed by looking at the Poincaré surface.

In a similar manner we verify that the region around $\omega = 15\frac{\text{rad}}{\text{s}}$ in Figure 3.8(c) has chaotic structure. Once again we plot the maximum Lyapunov exponent while varying ω to obtain Figure 3.15. It can be clearly sees that bifurcations occur when the Lyapunov exponent grazes zero, but more importunately, the region $14 < \omega < 15$ is indeed sensitive to initial conditions (positive Lyapunov exponent). This is a strong numerical evidence of chaos at this region.
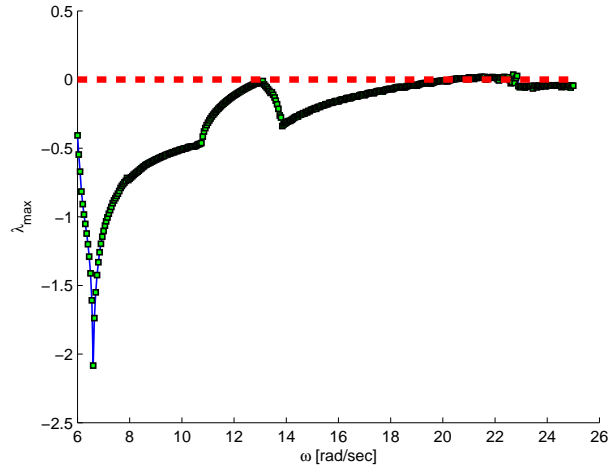
Figure 3.14: Largest Lyapunov exponents plot while varying ω for leg inertia $I_1 = 1 \cdot 10^{-3}$. The region where the Lyapunov exponents graze zero indicate bifurcation or quasiperiodic motion. Since the Lyapunov exponents are not positive they are not highly sensitive to initial conditions and do not infer chaotic region.
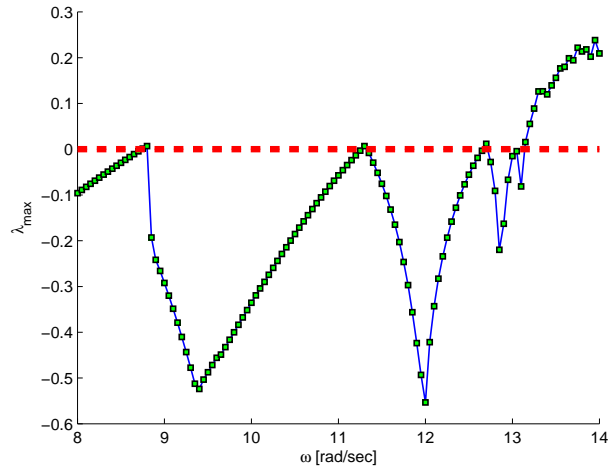
Figure 3.15: Largest Lyapunov exponents plot while varying ω for leg inertia $I_1 = 2 \cdot 10^{-4}$. The rightmost area where the Lyapunov exponents are positive infers sensitivity to initial conditions and a strong numerical evidence of chaos.

3.3.2 Efficiency

Using the efficiency calculation from obtained in Section 3.2.5 we plot Figure 3.16 showing the approximated efficiency while varying ω ((a)) and varying γ ((b)) using the same parameters as in the previous results Figure 3.9(c) and Figure 3.11(b).
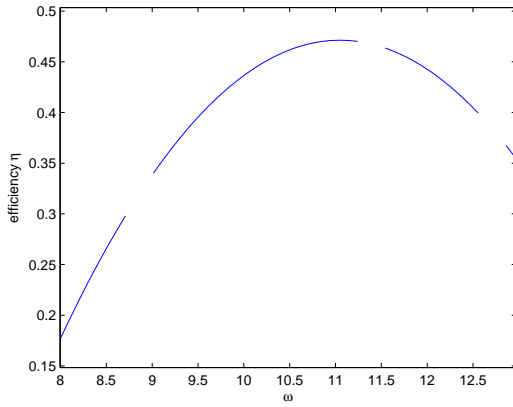
(a) Varying ω (b) Varying γ
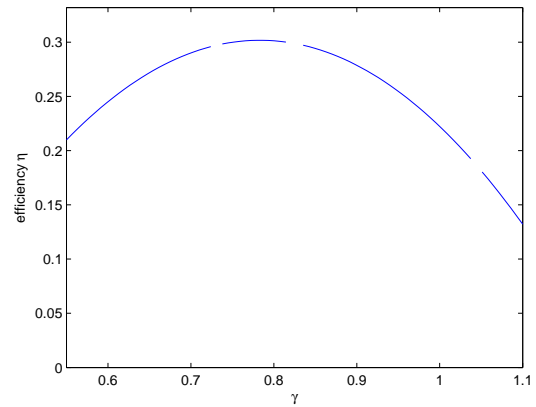
Figure 3.16: Efficiency calculation with change of input frequency ω for parameters as in Figure 3.9(b) ((a)), and CoM location γ for parameters as in Figure 3.11(b) ((b)).

71

(a) $\omega = 10\frac{\text{rad}}{\text{s}}$ (b) $\omega = 15\frac{\text{rad}}{\text{s}}$
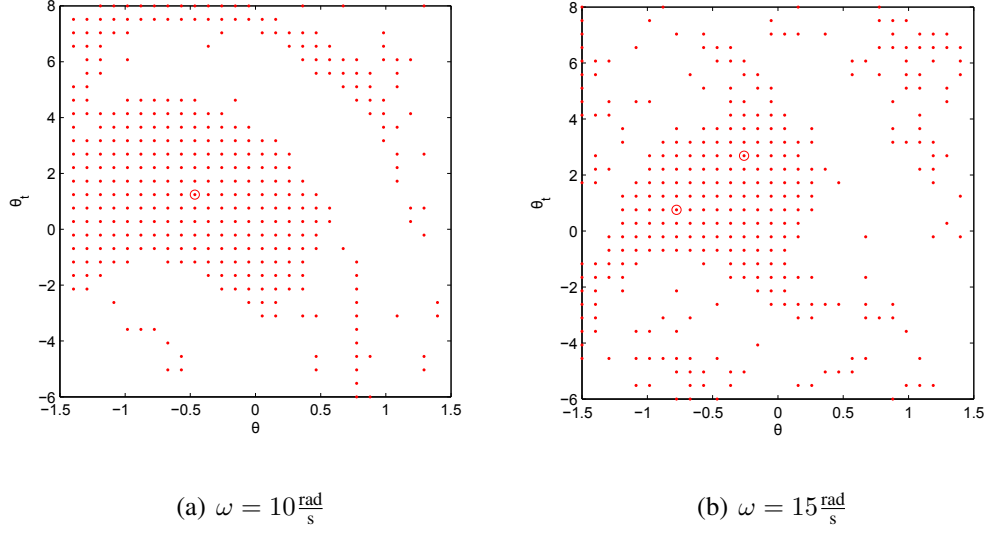
Figure 3.17: Basins of attractions using the cell mapping technique of two different input frequencies. Small dots representing the basin of attraction. Large dots represents attractor: (a) period-1 attractor, and (b) period-2 attractor.

3.3.3 Basin-of-attraction

Figure 3.17 shows two typical basins of attractions sampled from the period doubling cascade (Figure 3.9(a)) at two different input frequencies ($\omega = 10\frac{\text{rad}}{\text{s}}$, and $\omega = 15\frac{\text{rad}}{\text{s}}$). Figure 3.18 shows the basins of attractions sampled from the period doubling cascade (Figure 3.11(c)) at two different CoM locations ($\gamma = 1.1$, and $\gamma = 1$). The basin of attraction calculations were done using the modified simple cell mapping method described in Section 3.2.6 using a 30×30 grid. As mentioned earlier these basin of attraction results assume that the range of θ and $\dot{\theta}$ is not constrained, however, due to slippage, the actual mechanism's range is $\theta \in (0 : 1.2\text{rad})$, and $\dot{\theta} \in (-2 : 2\frac{\text{rad}}{\text{s}})$. Within this range the basin of attraction spans most, if not all the range in all four of these examples. These results demonstrate the robustness of this mechanism.

Approximation the basins of attraction is computationally intensive. A different method which approximates the stability is the disturbance rejection (Hobbelen and Wisse, 2007). Initial analysis using this method was used and has shown a relatively good correlation to the basin

(a) $\gamma = 1.1$ (b) $\gamma = 1$
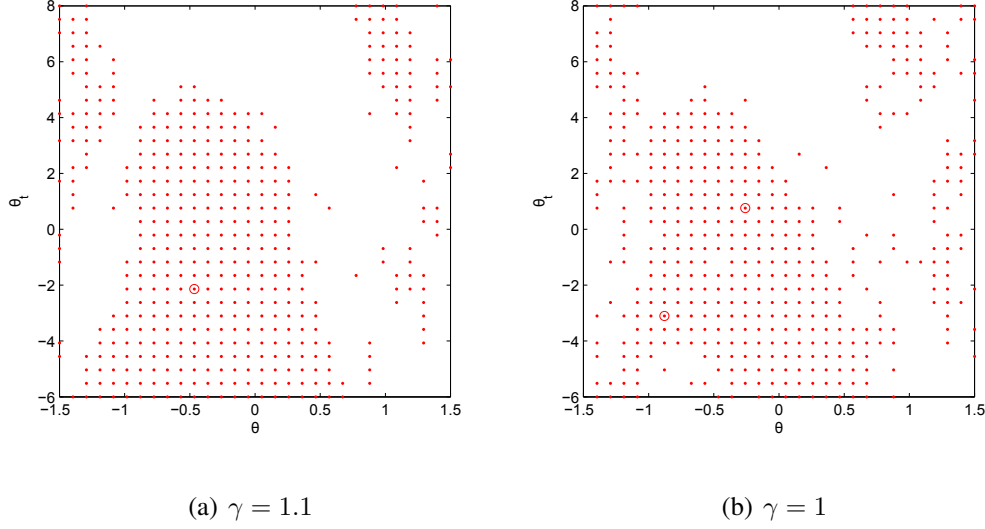
Figure 3.18: Basins of attractions using the cell mapping technique of two different CoM locations (γ). Small dots representing the basin of attraction. Large dots represents attractor: (a) period-1 attractor, and (b) period-2 attractor.

of attraction approximation. As future work, we will further use different and more efficient methods to approximate the stability.

3.3.4 Climbing rates

A practical measure of the ability to climb is to measure how far a mechanism climbs during each leap. This corresponds to one Poincaré map. However since asymmetric climbing occurs after the bifurcation points, a better measure might be the average climbing rate, i.e., $\Delta y = \sum_{k=1}^{N} \frac{\Delta y_k}{N}$, where N is the order of the period and Δy_k is the vertical distance of leap k. Figure 3.19 shows this average leap while varying ω ((a)) and varying γ ((b)) using the same parameters as in the previous results Figure 3.9(c) and Figure 3.11(b). While varying ω a continuous increase in climbing rate is shown as the input frequency is increased. More interestingly while varying γ, a noticeable increase in climbing rate occurs after the bifurcation from period-1 to period-2 ($\gamma \approx 1.07$. Apparently, after the period-2 bifurcation, the map initiating at large θ angle climbs
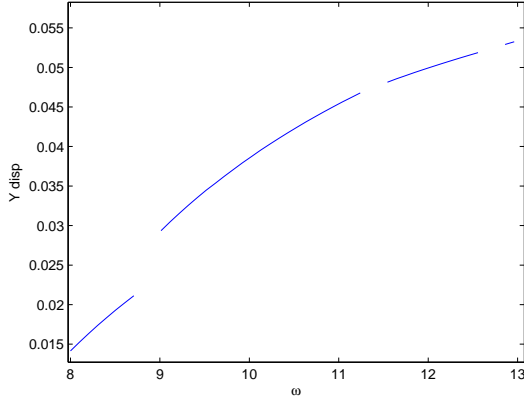
(a) Varying ω　　　　　(b) Varying γ
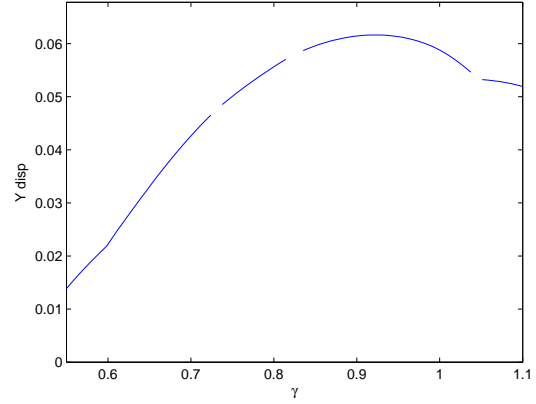
Figure 3.19: Averaged climbing rate with (a) change of input frequency (ω) and (b) CoM location (γ) .

significantly more than the period-1 map. Note that period-2 means a leap from one wall with a small θ angle followed by a leap with a large θ angle.

3.3.5　Numerical accuracy

For our numerical investigation we use Matlab$^{\text{TM}}$. For the numerical integration of the equations of motion we use ODE45 with an event-driven scheme to detect wall impact using tolerances of $1 \cdot 10^{-7}$. A multi-dimensional Newton-Raphson algorithm is used to find fixed points of the Poincaré map. The Jacobian of the linearized Poincaré map is calculated numerically using the central difference method with perturbation of size $1 \cdot 10^{-8}$. In order to decrease numerical errors in bifurcation plots we first make sure that the transient dynamics have decayed by forward simulating a few hundred Poincaré maps. In order choose an adequate integration step size, we have verified that the linear natural frequencies of the system are smaller than the lowest distinct value in the forced vibration power spectra.

3.3.6 Discussion: is limping better?

Figure 3.20 shows the maximum eigenvalue (characteristic multiplier), efficiency approximation, and climbing rate of the DSAC mechanism with same parameters of the ones shown in Figure 3.9(c) and Figure 3.11(b). The plots on the left of Figure 3.20 are for varying ω with leg inertia of $2 \cdot 10^{-4}\,\mathrm{kg\,m^2}$. The plots on the right are for varying γ with leg inertia of $3 \cdot 10^{-4}\,\mathrm{kg\,m^2}$. Light gray shade represent period-2 and darker shade represents period-4 gaits. As can be seen in the plots, the non-symmetric period-2 and period-4 have smaller magnitude eigenvalues corresponding to higher local stability[1], more efficient and higher climbing rates. In experiments shown in the next section, we were not able to recreate this phenomenon, most likely due to slippage at high angles at impact of the "limping" gaits.

---

[1]In fact, smaller eigenvalues suggest faster convergence from a perturbation back to the limit cycle

(a) Varying ω - Max Lyapunov Exp.

(b) Varying γ - Max Lyapunov Exp.

(c) Varying ω - calculated efficiency

(d) Varying γ - calculated efficiency

(e) Varying ω - climbing rate

(f) Varying γ - climbing rate
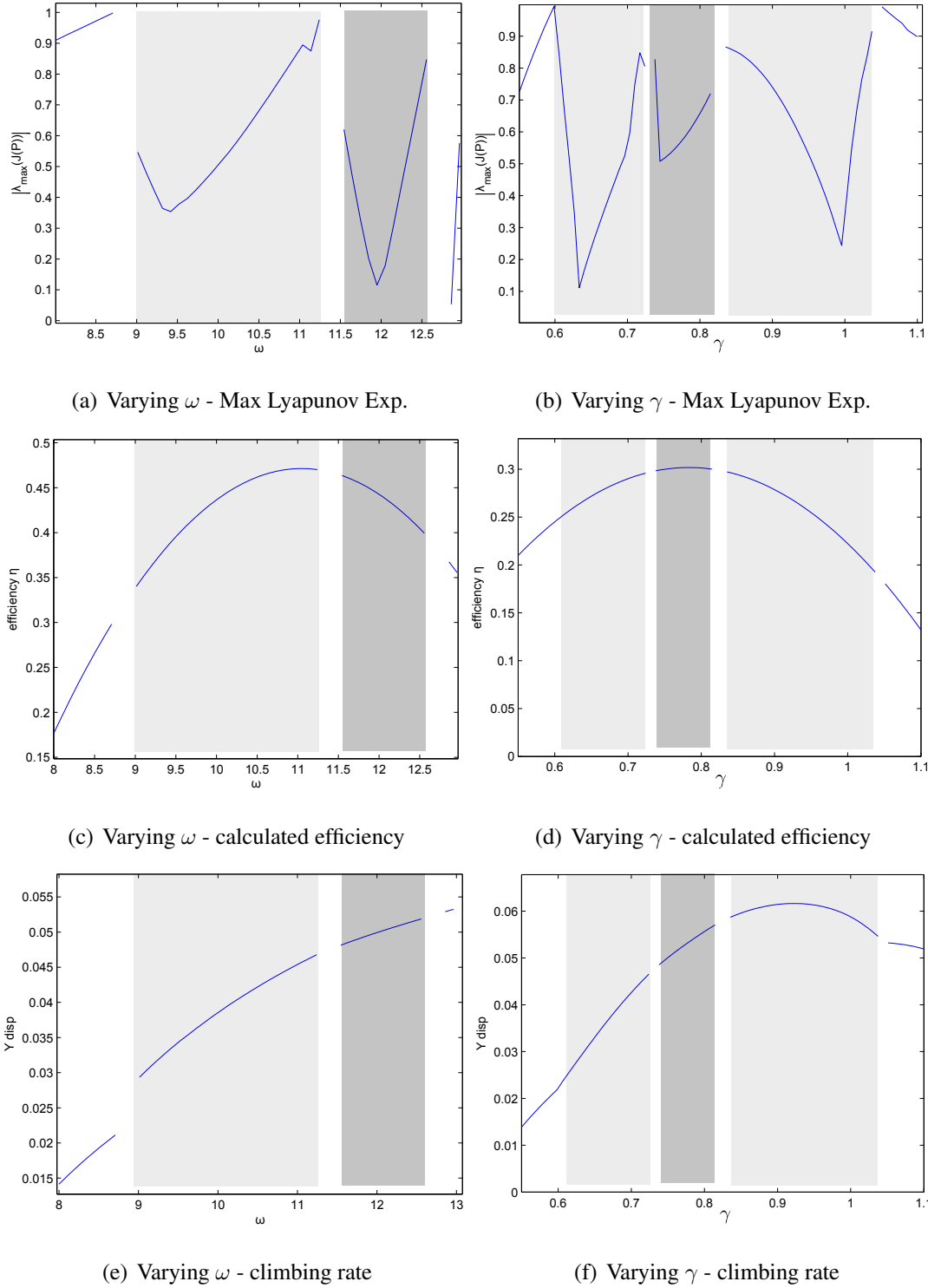
Figure 3.20: Is limping (period-2) better? Light gray shade marks regions of period-2 while darker shade marks period-4 regions. As can be seen in the plots, the non-symmetric period-2 and period-4 have lower Lyapunov exponent, more efficient and higher climbing rates.
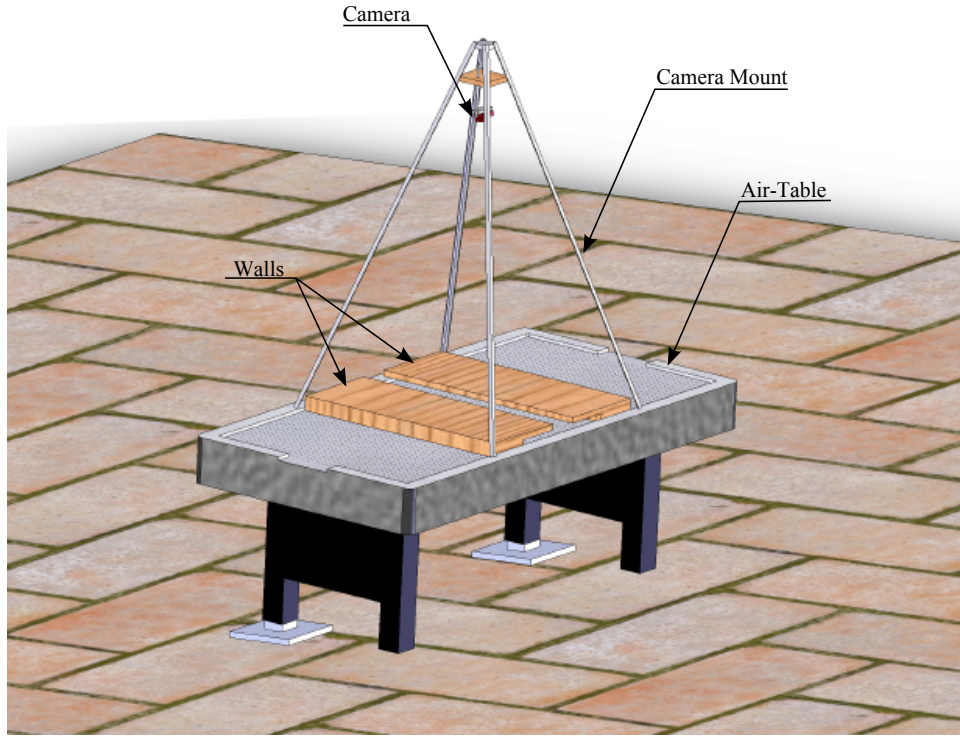
Figure 3.21: Air-table and tracking system mounted above.

3.4 DSAC Experiments

3.4.1 Experimental setup

The experimental setup consists of an air table which reduces the out-of-plane motions. The air table also allows to lower the effective gravity by inclining the table. As discussed earlier, we use a tenth of the normal gravitational acceleration. In order and track the mechanism, the Optitrack optical tracking system by (NaturalPoint^{TM}, 2009) is used. This system tracks passive IR markers at rates of 100[Hz]. Since only 2-D motions are needed to be tracked, we can use a single camera mounted normal to the surface of the air-table (see Figure 3.21).

Our current DSAC prototype design, shown in Figure 3.22, consists of a disk which increases the surface area between the mechanism and the air-table. On top of the disk the body mass is connected. On this same disk, a servo motor is connected to a light weight leg. An Arduino
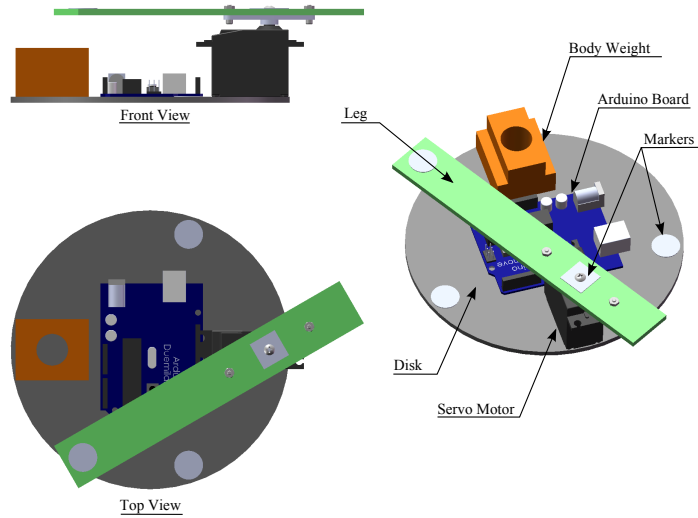
Figure 3.22: Current mechanism design.

microcontroller is in charge of the low level control of the leg. By using time interrupts, the microcontroller sends angle commands at exact timed events. This is used to produce a sinusoidal motion of the leg. The next section shows the verification of the sinusoid leg angle. A wireless connection is established through an XBEE wireless module between the microcontroller and the central PC. The PC is used to log data from the tracking system and send new control commands to the Arduino. In the experiments in this chapter the controls remain constant during the entire experiment. The duration of communication from the PC to the microcontroller (with a confirmation back to the PC) is about 50ms.

We show two different experiments for two different parameter sets. The approximated dimensional and nondimensional parameters for both experiments are given in Table 3.2. The first experiment, in section 3.4.3 describes a symmetric period-1 experiment. Section 3.4.4 describes a period-2 climbing gait with a different leg leg ratio (γ). We will also show that these results correlate well to simulation and to the period doubling results of section 3.3.

Table 3.2: Approximated DSAC parameters for experiment section

Dimensional Parameters			
Parameter	Description	Value in period-1 exp.	Value in period-2 exp.
m_1	leg mass	0.09 kg	identical
m_2	body mass	0.3 kg	identical
l_1	leg length	0.121 m	0.135 m
b_1	CoM of leg	0.115 m	0.13 m
b_2	CoM of body	0.1 m	identical
I_1	leg inertia	$2.5 \cdot 10^{-3}\,\mathrm{kg\,m^2}$	identical
I_2	body inertia	$1 \cdot 10^{-8}\,\mathrm{kg\,m^2}$	identical
d_wall	half wall width	0.08 m	identical
g	gravitational acceleration	$0.9807\,\frac{\mathrm{m}}{\mathrm{s}^2}$	identical
ω	sinusoid frequency	$17.3\,\frac{\mathrm{rad}}{\mathrm{s}}$	$16.5\,\frac{\mathrm{rad}}{\mathrm{s}}$
Nondimensional Parameters			
μ	Mass ratio $\frac{m_2}{m_1}$	$3\frac{1}{3}$	identical
β	CoM location of leg $\frac{b_1}{l_1}$	0.95	0.963
γ	link length ratio $\frac{b_2}{b_1}$	0.869	0.769
δ	leg to wall gap ratio $\frac{l_1}{d_\mathrm{wall}}$	1.51	1.687
ρ_1	nondimens. inertia $\frac{I_1}{d_\mathrm{wall}^2(m_1+m_2)}$	1.002	identical
ρ_2	nondimens. inertia $\frac{I_2}{d_\mathrm{wall}^2(m_1+m_2)}$	$4 \cdot 10^{-6}$	identical
Ω	nondimens. gravity $\frac{g}{\omega^2 d_\mathrm{wall}}$	4.94	4.712
A	sinusoid amplitude	0.28 rad	0.27 rad

3.4.2   Verifying ϕ angle assumption

In order to constrain the angle between the two links (ϕ) as a sinusoid, the microcontroller forces the servo to reach the maximum amplitude. Using time-based interrupts, the microcontroller then forces the servo to change direction and reach the minimum amplitude. The time between interrupts induces the frequency of the sinusoid. The method in theory should force a square wave, however, the motor dynamics causes a wave very close to a sinusoid. The method only works for a bounded region of A and ω, since at high ω and/or high amplitude, ϕ will not reach the prescribed angle and will cause a non symmetric sinusoid. Figure 3.23 shows the validation experiments preformed to verify the sinusoid assumption on ϕ. The experiments were performed by sending the microcontroller different A,ω inputs, and logging the actual ϕ angle outcome. These experiments were not performed during climbing motion, but on a flat terrain without contact the walls. In order to validate the sinusoid, a mean square curve fitting method is used together with validating that the variance of the peaks is low. White blocks in Figure 3.23 represent feasible sinusoids, and black blocks represent non-feasible parameters. These areas are the outcome of the servo not being fast enough to reach the specified sinusoid parameters. As example, three valid sinusoids are shown on the left and three non-valid sinusoids, where the servo did not reach the desired amplitude, are shown on the right. These non-valid sinusoids are the result of large amplitude to frequency ratio causing the motor to be non-symmetric. We use this discretized controls space when choosing sinusoid parameters for the experiments. As will be seen in the experiments in the following sections, even while impacting the walls, the ϕ angle assumption stays relatively correct, other than small perturbations during impacts.

80
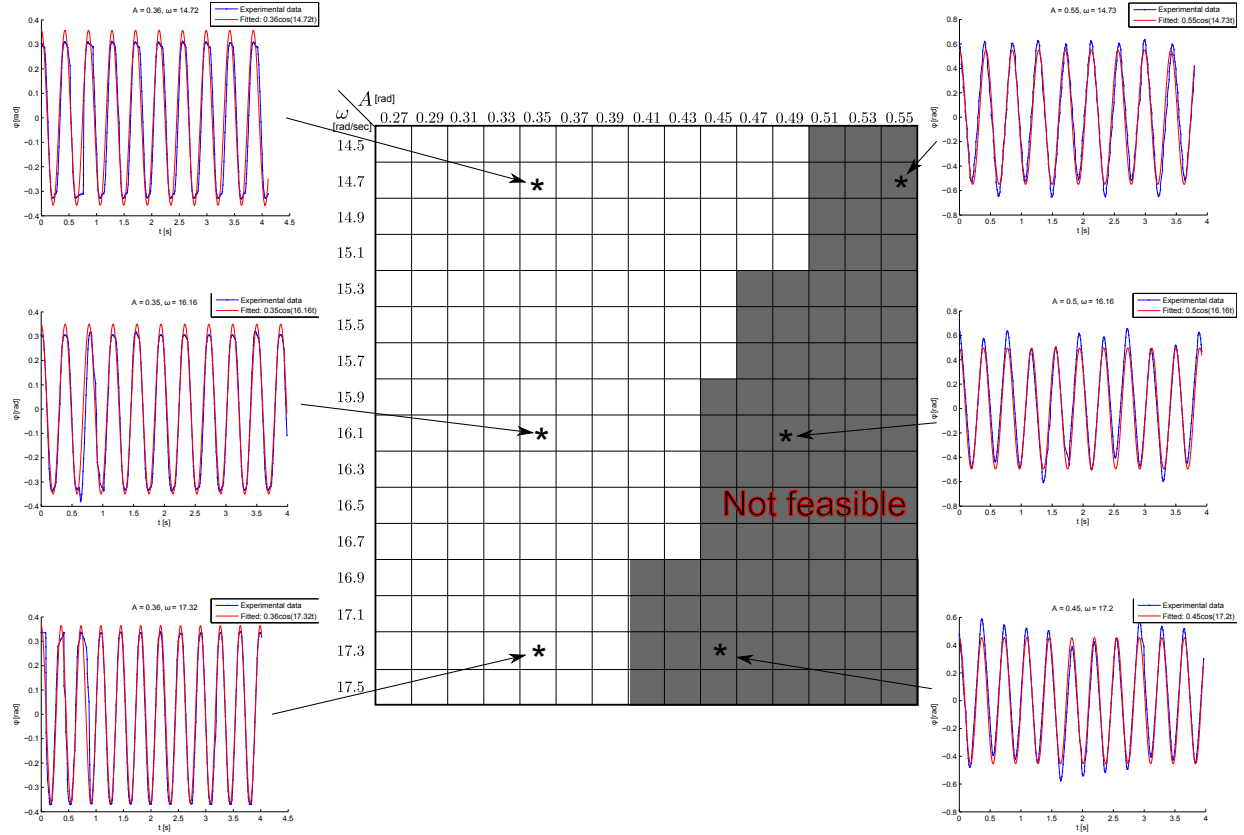
Figure 3.23: Verifying ϕ angle assumption. The plot shows a discretized controls space of different amplitudes (A) and frequencies (ω). White blocks represent feasible sinusoids where the parameters indeed formed a sinusoid. Black blocks represent non-feasible parameters. Three valid sinusoid plots are shown on the left and three non-valid sinusoid plots on the right.

3.4.3   Proof-of-concept experiments - Period-1

The following two sections consist of two experiments showing a bifurcation from period-1 to period-2 while varying the leg length ratio γ as was shown in experiments in Section 3.3.1.2. Figure 3.24 shows a sequence of images of the current design for $A = 0.28$rad, $\omega = 17.3\frac{\text{rad}}{\text{s}}$, and $\gamma = 0.869$. Since it is hard to distinguish between different periods by looking at these images, the corresponding configuration variable plots are given in Figure 3.25 and a closeup in Figure 3.26. These plots are obtained by tracking the four markers using the Optitrack system. As was assumed, ϕ is approximately sinusoid, other than small perturbations during impact. The plots of the leg angle θ include crosses that mark the points on the Poincaré surface including the flip after each impact. This flip, which negates the leg angle from impacting the left wall to the right wall was introduced in Section 3.1.2.2. This is used in order to compare leg angles on the left wall with leg angles on the right wall. Since these crosses overlap, we can confirm that for this leg length ratio, period-1 exists.

As a comparison between the experiment and the simulation from the analysis section we refer to Figure 3.27. This figure depicts simulation with the same mechanism parameters given in Table 3.2. Since our experimental prototype has a 1in thick leg and our simulation model uses a leg with no thickness, to compare the two, we change the amplitude and wall width in simulation to $A = 0.65$rad, $d_{\text{wall}} = 0.06$m. Qualitative, there is a good correlation between simulation and experiments. Both, simulation and experiment show a symmetric, period-1 climbing. The simulation differ mostly in the climbing rate. In simulations the robot advanced about twice as much as in the experiments.

Figure 3.24: DSAC Period-1 experiment - $\omega = 17.3\frac{\text{rad}}{\text{s}}$, $A = 0.28\text{rad}$ and $\gamma = 0.869$ – image sequence approximately 10ms apart.

Figure 3.25: DSAC period-1 experiment: $\omega = 17.3\frac{\text{rad}}{\text{s}}$, $A = 0.28$rad and $\gamma = 0.869$ – plot of configuration variables. Crosses mark the points on the Poincaré section, including the flip. The entire experiment consist of three climbs from $y = 0$m to $y = 0.7$m. The robot was manually taken down to $y = 0$m at the end of each run. A closeup of the last third of the experiment is given in Figure 3.26.

Figure 3.26: Closeup of of DSAC period-1 experiment: $\omega = 17.3\frac{\text{rad}}{\text{s}}$, $A = 0.28$rad and $\gamma = 0.869$ – plot of configuration variables. Crosses mark the points on the Poincaré section, including the flip. The plot of the leg angle θ reveals a symmetric period-1 climbing pattern. The plot of the angle ϕ, between to the two links follows the desired sinusoid.

Figure 3.27: Simulation comparing to DSAC period-1 experiment: $\omega = 17.3\frac{\text{rad}}{\text{s}}$, $A = 0.28$rad and $\gamma = 0.869$ – plot of configuration variables. As in the experiment plot, the simulation reveal a symmetric period-1 climb. Simulated climbing rate is approximately twice as fast as in experiment.

3.4.4 Proof-of-concept experiments - Period-2

The next experiment we performed shows a period-2 climbing motion for different leg length ratio (γ). Figure 3.28 shows a sequence of images of the current design for $A = 0.27$rad and $\omega = 16.5\frac{\text{rad}}{\text{s}}$, and $\gamma = 0.769$. Once again, since it is hard to distinguish between different periods by looking at these images, the corresponding configuration variable plots are given in Figure 3.29 and a closeup in Figure 3.30. Since these crosses on the θ plot, corresponding to the leg angles at the Poincaré sections, do **not** overlap, the climbing motion is a non-symmetric, period-2 for this leg length ratio.

As a comparison between the experiment and the simulation from the analysis section we refer to Figure 3.31. Once again, to more accurately correlate the experiment to simulation, we change the amplitude and wall width in simulation to $A = 0.62$rad, $d_{\text{wall}} = 0.06$m. Qualitative, there is a good correlation between simulation and experiments. Both, simulation and experiment show a non-symmetric, period-2 climbing. The simulation differ mostly in two parts. First, the leg angle (θ) on the Poincaré section deviates by about 30%. The second deviation of the simulation from experiments is the climbing rate. In simulations the robot advanced about four times as much as in the experiments. We predict that this difference is mostly due to leg slipping at these large leg angles during impact.

Figure 3.28: DSAC Period-2 experiment: $\omega = 16.5\frac{\text{rad}}{\text{s}}$, $A = 0.27\text{rad}$ and $\gamma = 0.769$ – image sequence approximately 15ms apart.
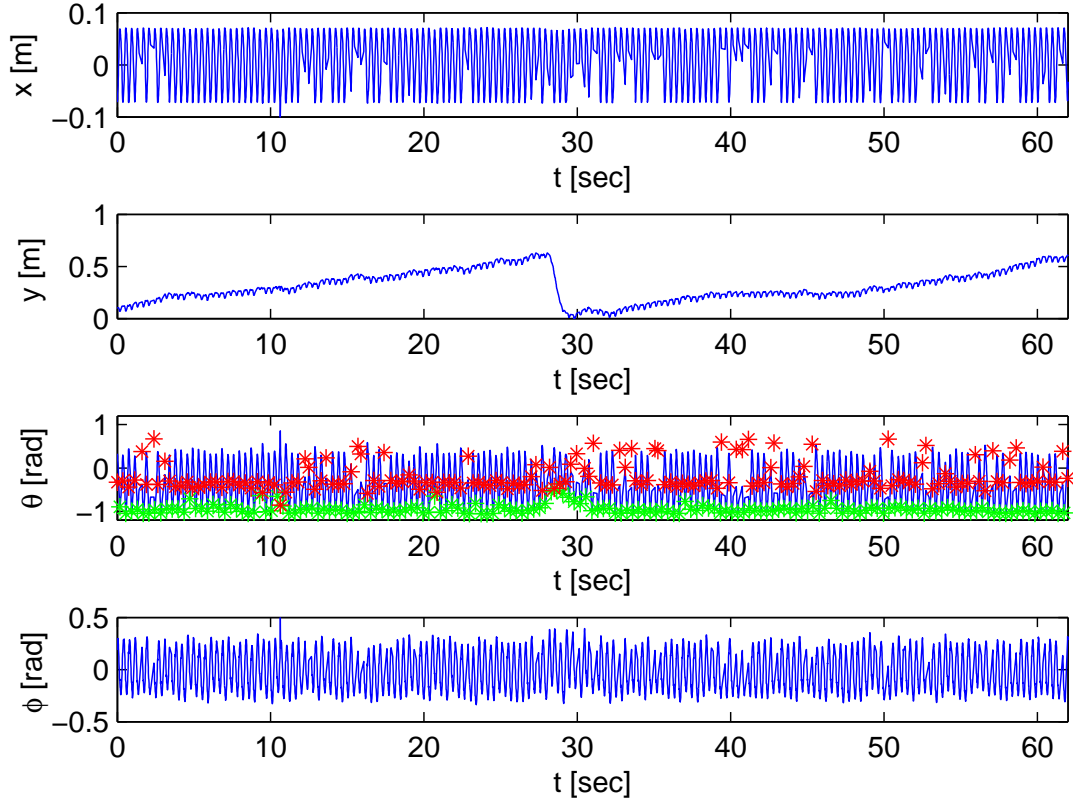
Figure 3.29: DSAC period-2 experiment: $\omega = 16.5\frac{\text{rad}}{\text{s}}$, $A = 0.27$rad and $\gamma = 0.769$ – plot of configuration variables. Crosses mark the points on the Poincaré section, including the flip. The entire experiment consist of two climbs from $y = 0$m to $y = 0.7$m. The robot was manually taken down to $y = 0$m at the end of each run. A closeup of 3.5sec-9sec of the experiment is given in Figure 3.30.
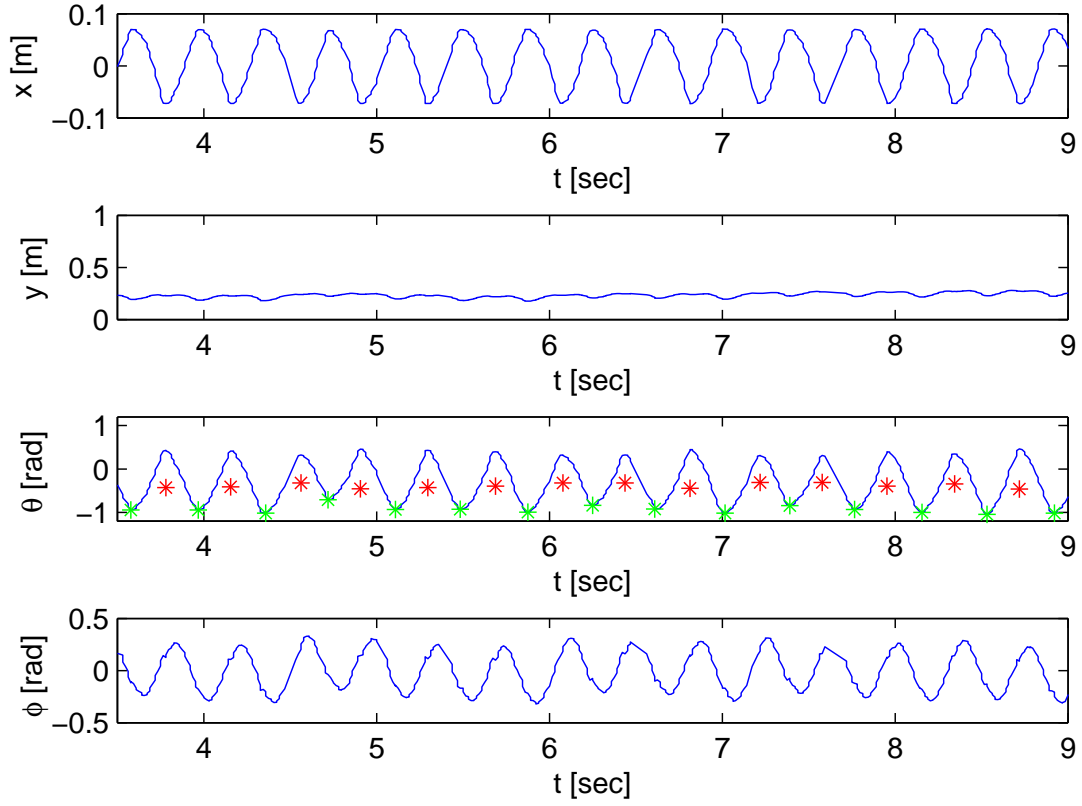
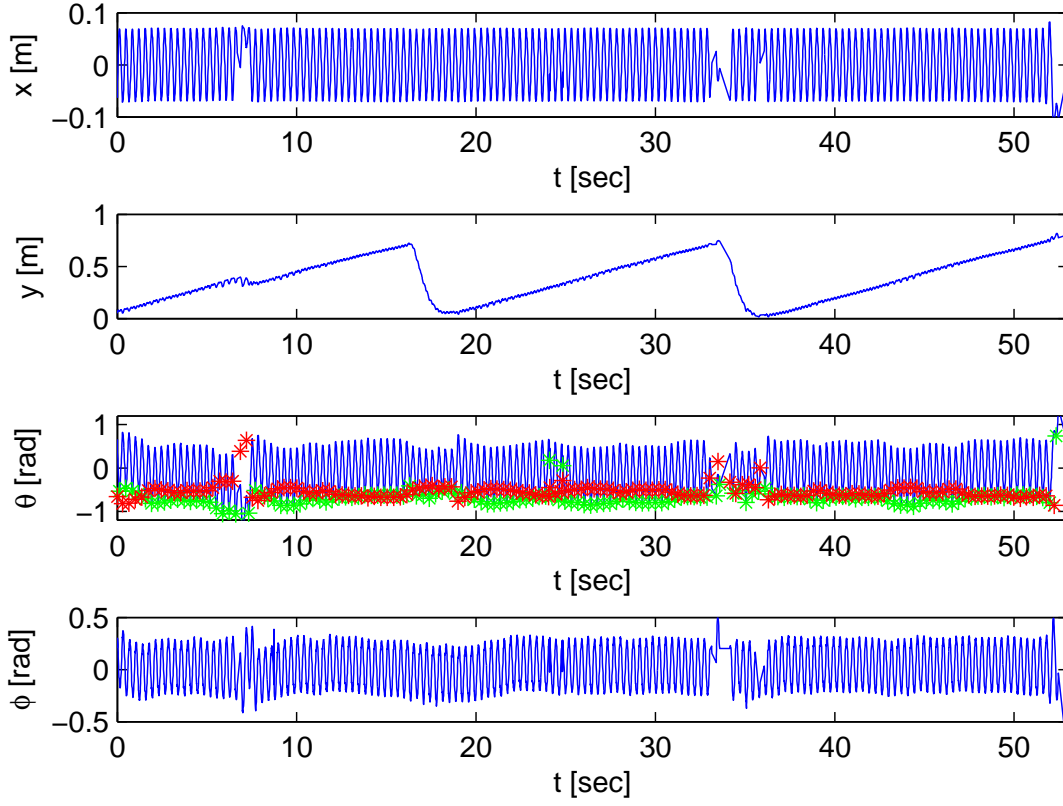Figure 3.30: Closeup of of DSAC period-1 experiment: $\omega = 16.5\frac{\text{rad}}{\text{s}}$, $A = 0.27$rad and $\gamma = 0.769$ – plot of configuration variables. Crosses mark the points on the Poincaré section, including the flip. The plot of the leg angle θ reveals a non-symmetric period-2 climbing pattern. The plot of the angle ϕ, between to the two links follows the desired sinusoid.

90

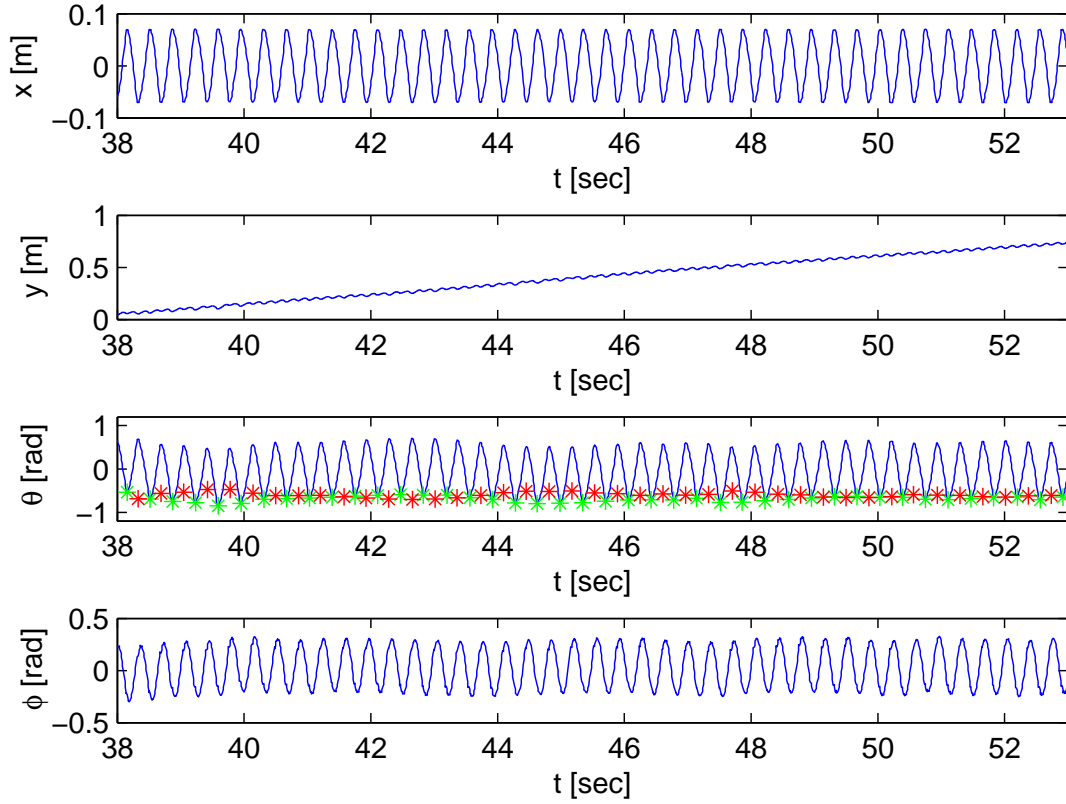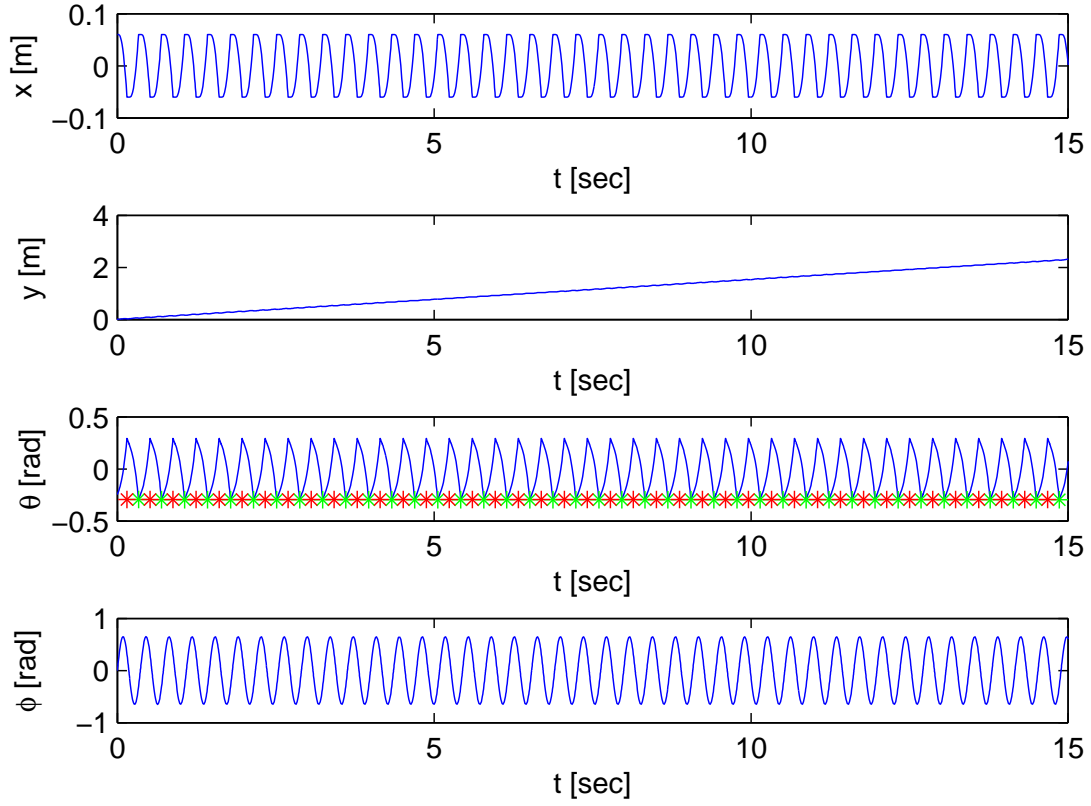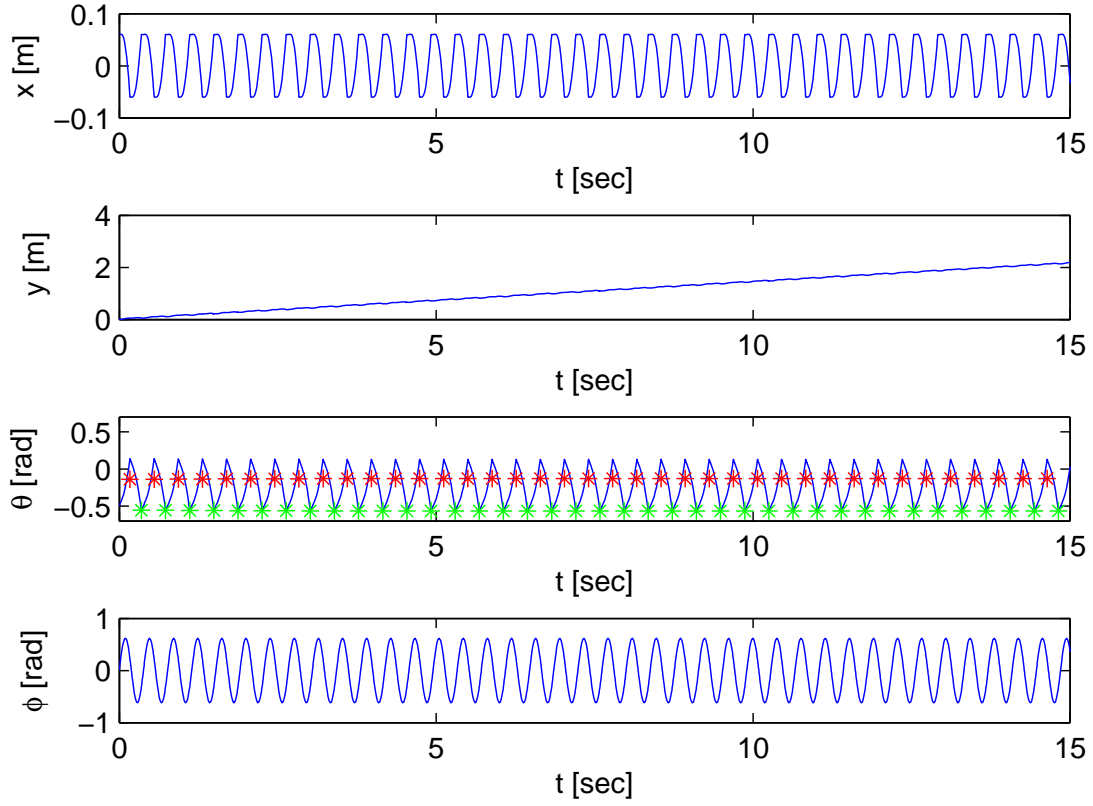Figure 3.31: Simulation comparing to DSAC period-2 experiment: $\omega = 16.5\frac{\text{rad}}{\text{s}}$, $A = 0.27\text{rad}$ and $\gamma = 0.769$ – plot of configuration variables. As in the experiment plot, the simulation reveal a non-symmetric period-2 climb. Simulated climbing rate is approximately four times as fast as in experiment.

Chapter 4

DTAR

4.1 Introducing DTAR

This chapter introduces a mechanism called DTAR for Dynamic, Tube Ascending Robot (depicted in Figure 4.1) which comprises only a body, a motor connected to an eccentric mass and two O-rings, can safely climb inside tubes with a single motor and no bristles. The motor rotates the eccentric mass with a constant angular velocity (ω). This unbalanced eccentric mass rocks the mechanism back and forth between the tube walls and propels the mechanism upwards.

The DTAR is an miniaturized extension of the DSAC described in Chapter 3. In addition to size, two main differences distinguish the current DTAR mechanism from the two link DSAC mechanism. The first difference is that DTAR can climb inside tubes. The second difference is that instead of using a motor which continuously changes direction in a symmetric oscillation, DTAR rotates an eccentric mass with a constant angular velocity.

Asymmetry in tube climbing mechanisms is crucial to direct the movement upward. The asymmetry in most climbing mechanisms is anisotropic friction such as the canted bristles described in (Gmiterko et al., 2002; Mištinas and Spruogis, 2002; Salomon et al., 2008). Unlike these mechanisms, the DTAR does not use anisotropic friction. This enables us to "turn off" the attachment mechanism just by stopping the motor rotation. In this chapter, we introduce DTAR

(a) CAD model   (b) Mechanism Schematics
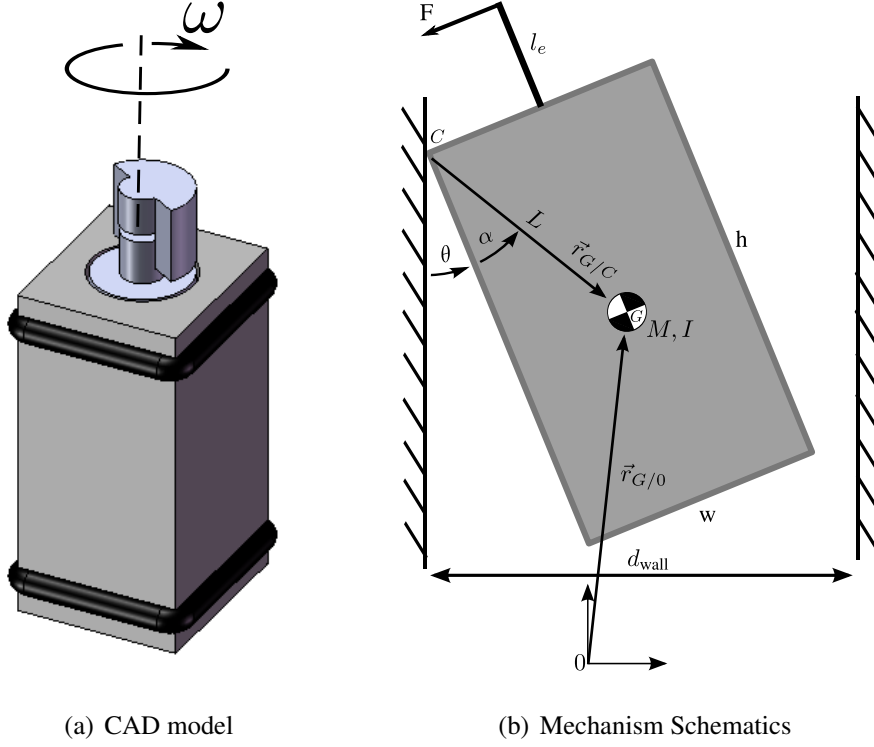
Figure 4.1: The DTAR mechanism; CAD image and planarized schematics.

and its gait. We then more carefully analyze two transitions in the gait to find lower bounds on the angular velocity needed to propel the mechanism stably upwards. We end with simulation results and proof-of-concept experiments showing the robustness of the DTAR.

4.2 Modeling and Analysis

The DTAR mechanism, as mentioned above, comprises a body, motor, eccentric mass and two O-rings. The motor rotates the eccentric mass at a constant angular velocity (ω). We model the excitation as an oscillating acceleration and force with amplitude $F = m\,\omega^2\,e$, where m is the eccentric mass and e is the eccentricity, i.e., the distance between the motor axis and the eccentric mass's Center of Mass (CoM).

In order to more easily analyze the behavior of this mechanism we planarize it by using a square cross section mechanism inside a square tube. This design restricts the mechanism to
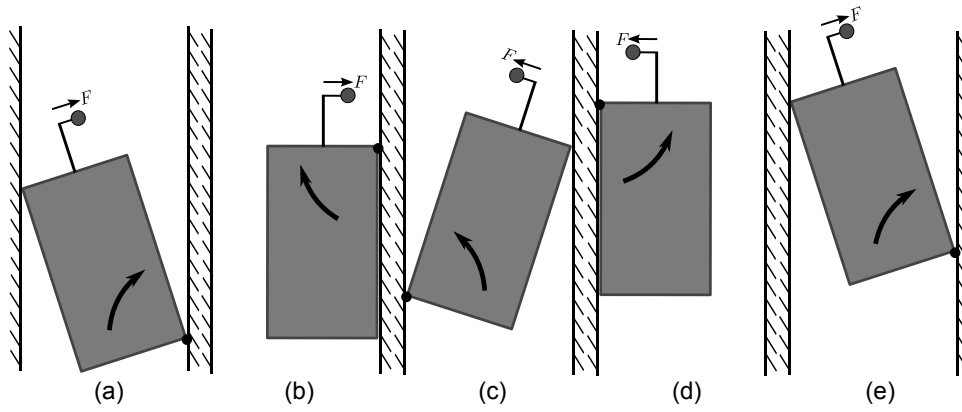
Figure 4.2: Gait sequence of DTAR from left to right. The oscillating force generated by the eccentric mass generates a torque around the contact points (O-rings), which in turn forms a stable periodic motion.

climb without twisting about the vertical axis. (A similar mechanism with circular cross section does climb inside a circular tube, while twisting about the vertical axis.) We use O-rings in order to increase friction, to accurately localize the contact points and to soften the impact with the walls.

The oscillating force generated by the eccentric mass generates a torque around the contact points, which in turn forms a stable periodic motion as can be seen in Figure 4.2. The first part of the gait sequence (Figure 4.2(a-c)) is mirrored in the last part of the sequence (Figure 4.2(c-e)). The following sections analyze two important conditions that must hold for stable climbing. We find a bound on the relationship between the input (ω) and the mechanism's parameters that will ensure this stable gait sequence. The first analysis examines the transition from Figure 4.2(b) to Figure 4.2(c). The eccentric mass's rotation must produce a torque about the rotation point high enough to ensure rotation to Figure 4.2(c). The second case examines the parameters that ensure jamming during climbing gate, i.e., the state where the mechanism contacts two points and does not slip. We conclude the analysis section by approximating the climbing rate of DTAR.

95

4.2.1   Rotating around contact point

The location of the eccentric mass is crucial to the stability and the direction of climbing. As mentioned earlier, to simplify the analysis we planarize DTAR (Figure 4.1(b)). Furthermore, we model the system as a single rigid body with externally applied forces, i.e., we assume that the eccentric mass generates the force F without changing the CoM of the whole mechanism[1]. A critical gait transition is from Figure 4.2(b) to Figure 4.2(c) and similarly from Figure 4.2(d) to Figure 4.2(e) where the torque about the upper contact point should be sufficient to rotate the body until the bottom contact point strikes the opposing wall.

To analyze the parameters required to move DTAR from Figure 4.2(d) to Figure 4.2(e), we equate the change of angular momentum to the sum of torques at the contact point c,

$$\dot{\vec{H}}_{/c} = \Sigma \vec{\tau}_{/c}, \tag{4.1}$$

where c is the location of the contact point, $\Sigma \vec{\tau}_{/c}$ is the sum of torques applied on the body, and $\dot{\vec{H}}_{/c}$ is the rate of change of the angular momentum about the contact point c. The rate of change of the angular momentum and the sum of torques are given by

$$\dot{\vec{H}}_{/c} = M\vec{r}_{G/c} \times \vec{a}_{G/0} + \ddot{\theta}I\hat{k} = \ddot{\theta}ML^2 + \ddot{\theta}I, \text{ and} \tag{4.2}$$

$$\Sigma \vec{\tau}_{/c} = \vec{r}_{G/c} \times (-Mg)\hat{j} + Fl_e = -MgL\sin(\theta + \alpha) + Fl_e, \tag{4.3}$$

where $\vec{r}_{G/c}$ is the vector from the contact point to the CoM of the main body, and $\vec{a}_{G/0}$ is the acceleration of the main body relative to the inertial frame. The magnitude of vector $\vec{r}_{G/c}$, L and the angle α are denoted in Figure 4.1(b). Inserting Eq. 4.2 and Eq. 4.3 into Eq. 4.1 we obtain

$$\ddot{\theta}(I + ML^2) = Fl_e - MgL\sin(\theta + \alpha). \tag{4.4}$$

which leads to

$$\ddot{\theta} = \frac{Fl_e - MgL\sin(\theta + \alpha)}{(I + ML^2)}. \tag{4.5}$$

---

[1]This is a valid approximation since the torque applied by the eccentric mass around the contact point is approximately 10 times larger than the one applied by the change of CoM of the eccentric mass.

In order to rotate around the contact point and transition from the pose in Figure 4.2(d) ($\theta = 0$) to the pose in Figure 4.2(e), the angular acceleration of the body, $\ddot{\theta}$, must be positive, which yields an approximate constraint on the excitation force

$$F > \frac{MgL\sin(\alpha)}{l_e}. \tag{4.6}$$

By assuming that the excitation force is $F = m\,\omega^2\,e$, we can derive the constraint on the input parameter ω

$$\boxed{\omega > \sqrt{\frac{MgL\sin(\alpha)}{m\,e\,l_e}}.} \tag{4.7}$$

4.2.2 Ensuring jamming

Another important state to analyze is jamming (Figure 4.2(a,c,e)) where the friction at the two contact points must prevent sliding. The forces acting on the body are the contact forces, gravity, and the force from the eccentric mass. We use the moment labeling scheme Mason (2001) as an intuitive way to represent the composite wrench cone of contact forces that can act on the mechanism by the walls (see Figure 4.3) and balance the resultant wrench of the eccentric mass force and the gravity. We first draw the friction cone at each of the two contact points. Next, we label all the points to the left of the friction cone as $(+)$ and to the right of the friction cone as $(-)$, as in Figure 4.3(b) and Figure 4.3(c). We then keep all the regions with consistent labels as in Figure 4.3(d). Finally, the resultant wrench w_r, comprising the force from the eccentric mass rotation and the gravitational force, can be balanced by a wrench from the contact forces only if the resultant wrench passes to the left of the $(+)$ region, as depicted in Figure 4.3(d)[2].

Using the moment labeling method we found the $(+)$ region through which the resultant wrench should not pass. We next find the analytic expression for the critical parameter, so that the resultant wrench does not pass through this region. For practical choices of DTAR's parameters,

---

[2]In our case the only consistent label are the $(+)$ region, but in general $(-)$ regions can exist, and the resultant wrench must pass to the right of the $(-)$ in order to be balanced.
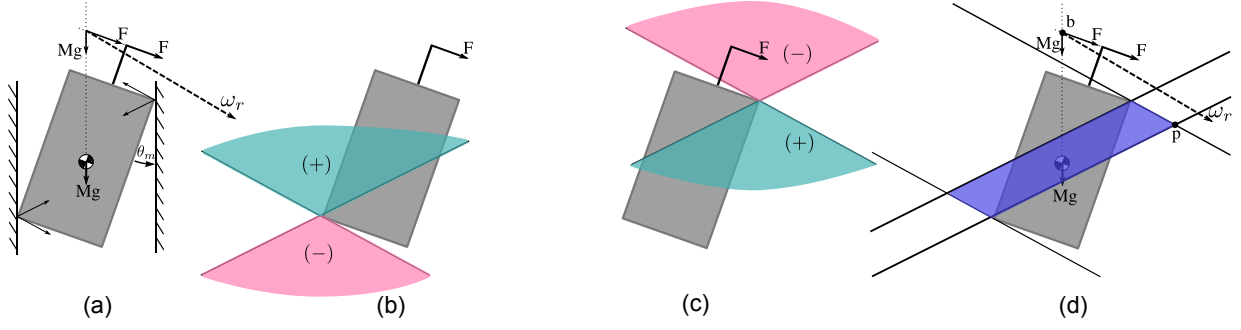
Figure 4.3: Ensuring jamming using the moment labeling method. (a) The contact friction cones are shown together with the resultant wrench w_r composed of mg and F. Figures (b) and (c) represent the nonnegative linear combination of the two lines of forces of the friction cone by moment labels. (d) The moment labeling representation of the composite wrench cone of contact forces the wall can apply to the mechanism. In this configuration, the resultant wrench w_r passes to the left of the $(+)$ set, implying that this resultant wrench can be resisted by forces within the composite wrench cone of the contact forces and hence will stay jammed.

the critical vertex is the top right corner of the $(+)$ region. We note that in extreme cases when l_e is small or μ is very large, the critical vertex might be the top left.

In order for the wrench w_r not to pass inside the $(+)$ region, the force (F) applied by the eccentric mass must satisfy the inequality

$$F > Mg \sec \theta_m \cot \left(\frac{p_y - b_y}{p_x - b_x} \right),\tag{4.8}$$

where p_x and p_y, the coordinates of the top right corner of the $(+)$ region in Figure 4.3(d), are

$$p_x = \frac{1}{2\mu} \left(\mathrm{h}\cos\theta_m - \mathrm{w}\sin\theta_m + \mu\mathrm{w}\cos\theta_m + \mathrm{h}\mu\sin\theta_m \right)$$

$$p_y = \frac{1}{2} \left(\mathrm{h}\cos\theta_m - \mathrm{w}\sin\theta_m + \mu\mathrm{w}\cos\theta_m + \mathrm{h}\mu\sin\theta_m \right),$$

the origin coordinates of the w_r vector, b_x and b_y are

$$b_x = \frac{1}{2} \left(\mathrm{w}\cos\theta_m + \mathrm{h}\sin\theta_m \right)$$

$$b_y = \left(\mathrm{h} + l_e - \theta_m\mathrm{w} \right)\cos\theta_m + \frac{1}{2} \left(\mathrm{h}\theta_m - 2l_e\theta_m - \mathrm{w} \right)\sin\theta_m,$$

98
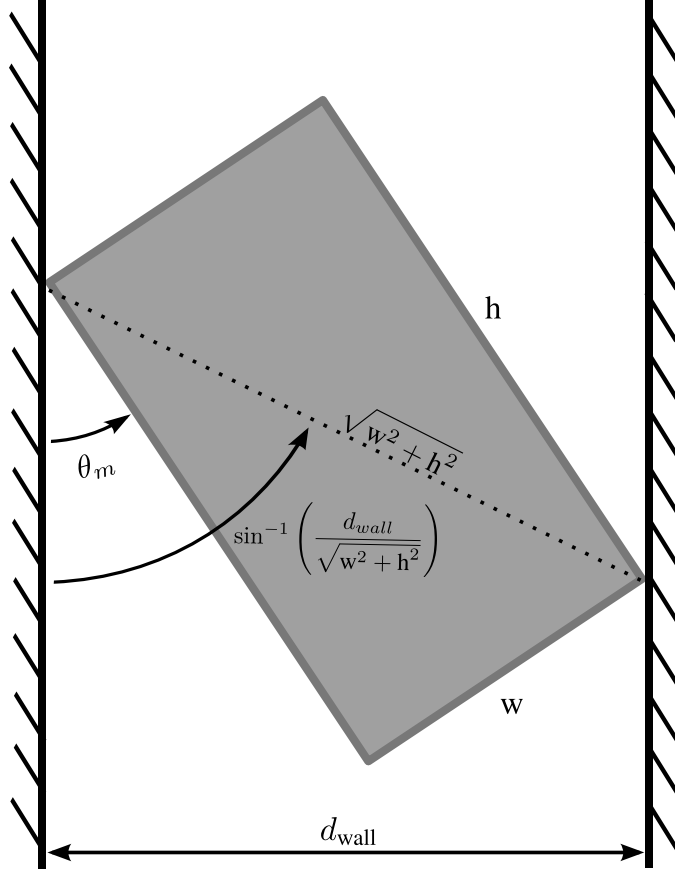
Figure 4.4: Schematics of DTAR in jamming state where $\theta = \theta_m$ is at maximum, such as in Figure 4.2(a,c).

and the angle in the jamming state, θ_m, such as in Figure 4.4 is

$$\theta_m = \sin^{-1}\left(\frac{d_{\text{wall}}}{\sqrt{\text{w}^2 + \text{h}^2}}\right) - \tan^{-1}\left(\frac{\text{w}}{\text{h}}\right), \tag{4.9}$$

where w, and h are the width and height of the mechanism, respectively, and d_{wall} is the tube width.

Once again, by assuming that the excitation force is $F = m\,\omega^2\,e$, using Eq. 4.8, we can derive the constraint on the input parameter ω

$$\omega > \sqrt{\frac{Mg\sec\theta_m\cot\left(\frac{p_y - b_y}{p_x - b_x}\right)}{m\,e}}. \tag{4.10}$$

99

In order to ensure the rotation around the contact point and jamming, the angular frequency of the eccentric mass (ω) should exceed both inequalities of Eq. 4.7 and Eq. 4.10.

4.2.3 Climbing rate

To estimate the climbing gate we first assume that the first two conditions are satisfied, i.e., the angular frequency of the eccentric mass (ω) satisfies the inequalities in Eq. 4.7 and Eq. 4.10. Furthermore we assume that, as shown in Figure 4.2, at least one contact point exists during the gait cycle. Therefore, we can find the climbing rate by calculating the vertical increment in one climbing period from Figure 4.2(a)-(e).

The vertical increment of the bottom left corner of the main body from Figure 4.2(a) to Figure 4.2(b) is

$$\text{w} \sin \theta_m, \tag{4.11}$$

and the vertical increment of the bottom left corner from Figure 4.2(b) to Figure 4.2(c) is

$$\text{h} \left(1 - \cos \theta_m\right). \tag{4.12}$$

In one full cycle, these vertical increments are executed twice. Moreover, in order to find the climbing rate, the period of the rotating eccentric mass should be included. Therefore, the climbing rate is

$$V = \frac{\omega}{\pi} \left(\text{w} \sin \theta_m + \text{h} \left(1 - \cos \theta_m\right)\right). \tag{4.13}$$

4.3 Simulation

We have simulated and verified the motion and climbing rate of the mechanism using the planar simulator WorkingModel2D (Design Simulation Technologies, Inc). Six snapshots showing the importance of the location of the eccentric mass are depicted in Figure 4.5. The simulation

100
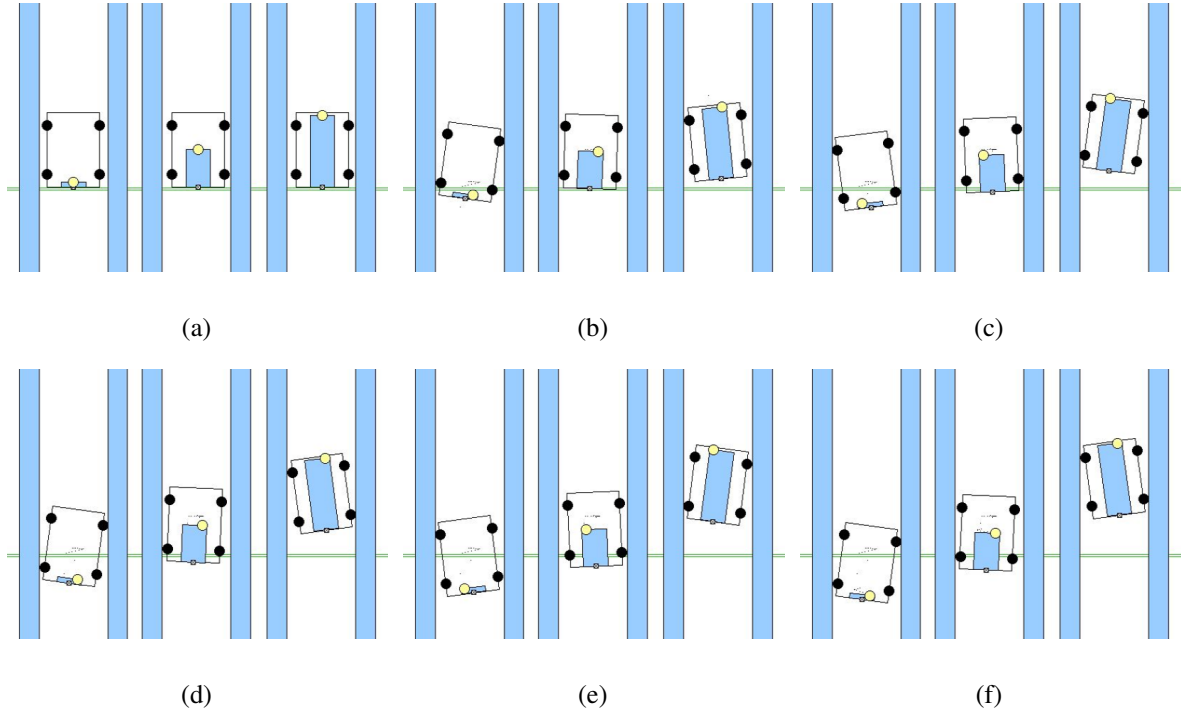
Figure 4.5: Simulation of three DTARs with different locations of the eccentric mass. From left to right, the eccentric mass is below the bottom O-ring, between the two O-rings, and above the top O-ring, respectively. The green line shows initial vertical placement of the three mechanisms. Simulation results verify the importance of the location of the eccentric mass.

depicts three different choices of eccentric mass location. When the eccentric mass is located below the bottom O-ring (leftmost mechanism) the climbing direction is downward. Locating the eccentric mass between the O-rings (middle mechanism) causes an unstable motion. Importantly, as shown in the analysis section, when the eccentric mass is located above the top O-ring DTAR stably climbs upwards.

4.4 Experiments

We have built a prototype of the DTAR using a small DC motor working at 5V, 32.1mA (no load). We have tested the DTAR in a few scenarios all climbing vertically upwards. We note
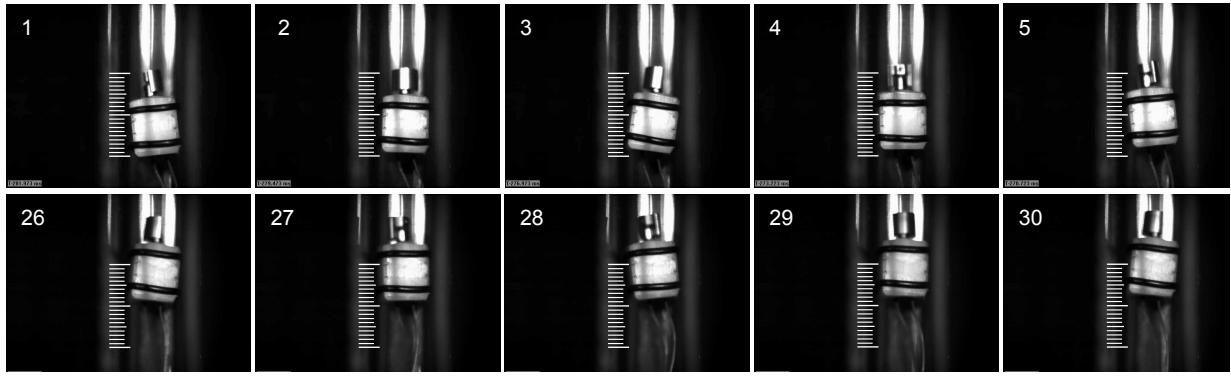
Figure 4.6: High speed footage of DTAR inside a square tube (starting top left). The first row shows the start and the second row shows the end. The whole sequence is approximately 100ms long, with snapshots separated by approximately 3ms. The total vertical displacement is approximately 14mm.

that the DTAR climbs just as well when the tube is tilted and not vertical. In fact, the angular frequency (ω) needed is lower since the gravity is reduced. As previously mentioned, when using a round cross section mechanism in a round tube DTAR naturally twists about the vertical axis. In order to isolate the climbing motion from the twisting motion, we have built a square cross section mechanism that climbs inside square tubes. The vertical velocity of DTAR can be up to 1/4 m/s, which correlates to about 20 body lengths per second. Moreover, in our experiments, the DTAR was able to carry approximately 20 grams (not at full speed) which is approximately five times its body weight. See the accompanying multimedia material. Parameters of the DSAC mechanism and the environment for the next sections are given in Table 4.1.

4.4.1 Straight square tube

Initially we tested the DTAR inside a straight square tube and recorded the experiment using a high speed camera (Figure 4.6). In this experiment the DTAR climbed vertically approximately 14mm in 100ms, equivalent to $0.14\frac{m}{s}$. Our previously derived estimate (Eq. 4.2.3) predicts a

Table 4.1: Approximate experimental parameters

Param.	Description	Value for straight square tube experiment (Sec. 4.4.1)	Value for varying width tubes experiment (Figure 4.7(a))	Value for S-shaped experiment (Figure 4.7(b))	Value for Y-shaped experiment (Figure 4.7(c))
w	Body width	0.013m	0.0128m	0.0128m	0.0128m
h	Body height †	0.008m	0.0154m	0.0154m	0.0154m
d_{wall}	Width of tube	0.0135m	0.014, 0.016, 0.018, 0.020m	0.018m	0.016m
μ	Friction coeff.	0.5	0.5	0.5	0.5
m	Eccentric mass	0.0005kg	0.0005kg	0.0005kg	0.0005kg
e	Eccentricity	0.0013m	0.0013m	0.0013m	0.0013m
M	Total mass	0.0025kg	0.0025kg	0.0025kg	0.0025kg
l_e	Shaft length	0.005m	0.005m	0.005m	0.005m
ω	Angular freq.	600rad/s	600rad/s	600rad/s	600rad/s
† body height is measured between O-rings					

climbing rate

$$V = \frac{\omega}{\pi}\left(\mathrm{w}\sin\theta_m + \mathrm{h}\left(1 - \cos\theta_m\right)\right) = 0.1673\left[\frac{\mathrm{m}}{\mathrm{s}}\right], \tag{4.14}$$

which is just over 15% error. Using the parameters in Table 4.1, we have verified that the minimum angular frequencies derived from Eq. 4.7 and Eq. 4.10 are indeed less than the one used in the experiment.

4.4.2   Varied terrain

To show DTAR's robustness we tested it in a variety of tube shapes including: (1) four square tubes varying in width in 2mm increments; (2) an S-shaped tube; (3) a Y-shaped junction (see Figure 4.7). Table 4.1 displays the DTAR and tube parameters for these experiments. The DTAR climbed well in all cases. In the narrowest tube (leftmost of Figure 4.7(a)) DTAR climbed very slowly since the clearance between the mechanism and the tube was small. In the S-shaped tube the tether slowed down the climbing rate. In the Y-shaped tube in Figure 4.7(c), We have no control authority over the selection of which branch to climb. In future work we intend to find a way to steer the mechanism.

Once again, using the parameters in Table 4.1, we have verified that the minimum angular frequencies derived from Eq. 4.7 and Eq. 4.10 are indeed less than the one used in the experiment. In Sec. 4.2.3, we approximated the climbing rate by assuming that the gait in Figure 4.2 is held. This assumption breaks and these approximations are not valid when wall width is relatively large in comparison to the width of the robot, such as in the two wide tubes in Figure 4.7(a). In the future, we intend to find a more general climbing rate approximation.
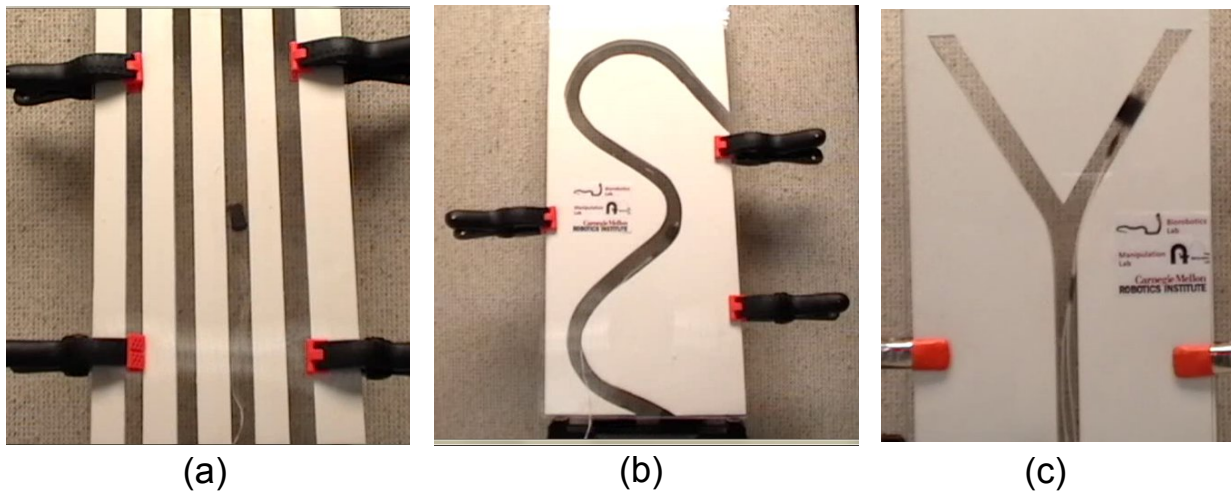
Figure 4.7: Various shaped tubes. a) variable width, b) S shaped tube, c) Y-shaped tube. In all cases DTAR climbed well. In the narrowest tube (leftmost of (a)) DTAR climbed very slowly since the clearance between the mechanism and the tube was small. In the S-shaped tube the tether slowed down the climbing rate.

Chapter 5

ParkourBot

The ParkourBot is inspired in part by the grace, efficiency, and adaptability of human *parkour*, sometimes known as free running. Parkour is the art of moving from place to place as quickly and efficiently as possible, overcoming obstacles using leaps, swings, rolls, and other dynamic movements. Whereas walls, chutes, and trenches are obstacles that may not be navigable using less dynamic forms of locomotion, in parkour these "obstacles" are potential sources of reaction forces for a well-placed hand or foot. For parkour practitioners ("traceurs") to make maximum use of these handholds and footholds, they must have precise control of their body dynamics. By taking advantage of dynamics, and by knowing the geometry and contact properties (e.g., friction and restitution) of the environment, the set of reachable states by parkour is vastly increased over that by more conventional locomotion.

To proceed efficiently over obstacles, the traceur stores energy elastically in muscles and tendons, and kinetically in translation and rotation. These energies can then be directed to move seamlessly from one task to the next. The climbing robot in this chapter, called ParkourBot, is based on two dynamic robots which we have previously built. The first, the BowLeg hopper (Brown and Zeglin, 1998; Zeglin, 1999), has high energy efficiency and requires feedback for stable hopping. The second, the DSAC described above, sacrifices energy efficiency for stable open-loop climbing of a chute. This chapter describes the design and control of a biped
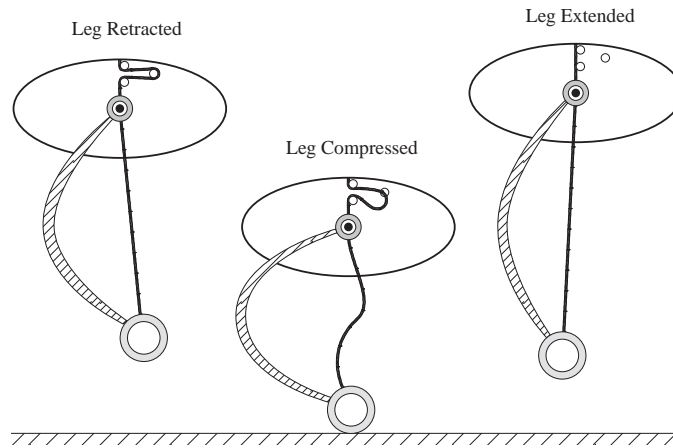
Figure 5.1: Schematics of the BowLeg monopod (reprinted with permission from Zeglin (1999)).

BowLeg climber (Figure 1.11) that inherits properties of the BowLeg hopper and the DSAC. The ParkourBot is designed to address different types of climbing tasks such as simple chutes and more complex environment having footholds at different orientations.

5.1 Mechanical Design

As mentioned previously, this current ParkourBot is based on the BowLeg hopper (Brown and Zeglin, 1998). However, in order to adapt it to a climbing scenario, we have decreased the size and have implemented a new mechanism design. This section will first explain the previous BowLeg concept and later review some of the new mechanism design components.

5.1.1 BowLeg design

The BowLeg (Brown et al., 2007), a robotic leg concept developed at Carnegie Mellon, integrates the functions of structure and spring into a lightweight leg. As shown in Figure 5.1, the BowLeg resembles an archer's bow, with the hip joint at one end and the foot at the other end of the bow. Made of unidirectional fiberglass ("bow glass"), the BowLeg can store a large amount of elastic energy, typically enough to lift its own weight 50 meters vertically. The concept of

108

a single-leg hopping machine using BowLeg technology is illustrated in Figure 5.1. When the BowLeg hopper is in flight, a low-power actuator stores energy in the BowLeg by tensioning a string attached to the foot. A separate actuator uses control strings to position the leg for the next impact. Upon impact, the string becomes slack and the BowLeg quickly releases its stored energy. The leg rotates freely about a hip joint, so that the foot matches ground speed without actuation, and no attitude-disturbing torque is transmitted through the joint. Hopping motion is controlled by choosing the angle of the leg at impact and the amount of energy stored in the BowLeg during flight. A special clutch mechanism (Figure 5.2) is used to release the leg when impact with the floor occurs.

5.1.2   ParkourBot design overview

The current ParkourBot depicted in Figure 5.3 comprises five primary parts: disk, BowLeg, leg angle control mechanism, retract mechanism, and a gyro-stabilizer. The disk, the main body of the robot, floats on top of an inclined air-table in order to planarize the system while enabling lower effective gravity. The BowLeg includes a thin, unidirectional fiberglass strip that provides the main leg structure and elastic energy storage; the hip lever that softens the leg stiffness and connects to the hip; the bow string that passes through the hip, connecting the foot to the retract mechanism; and a rubber foot pad that enhances foot traction. The remaining parts of the climber are described below.

5.1.3   Retract mechanism

The retract mechanism pulls on the bow string and stores potential energy in the springy BowLeg. It also includes a clutch to release the spring energy when the string goes slack during stance. As shown for the BowLeg Hopper in Figure 5.2, when the leg contacts the wall, the string slackens and the clutch disengages the string from the retract arm. This enables the BowLeg to fully extend and "kick" the wall. The initial design of the retract mechanism for the climber was a
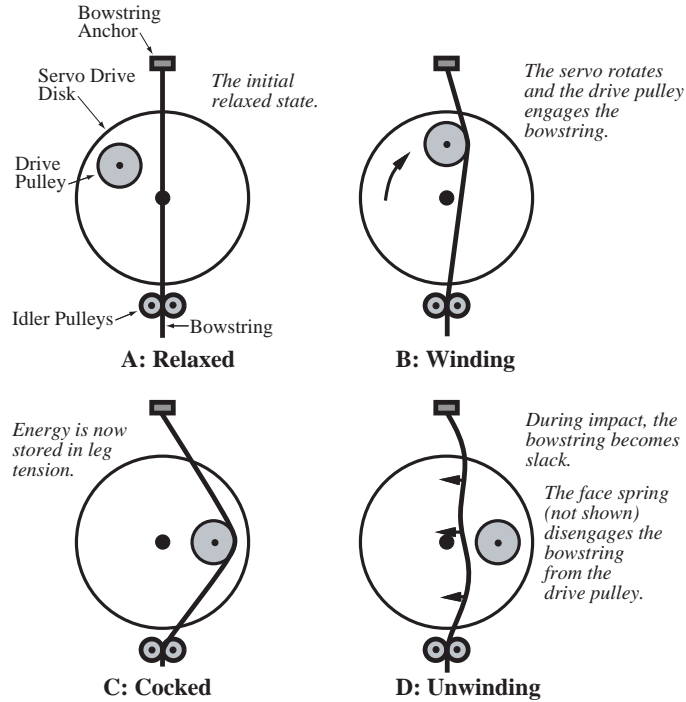
Figure 5.2: Schematic of the prototype thrust mechanism which stores energy in the leg during flight. The cycle begins in the relaxed state (A). During winding (B), the servo disk rotates, the drive pulley engages the bowstring, and the displacement of the bowstring compresses the leg (not shown). The energy stored in the cocked position (C) is a function of rotation angle. During the impact (D), the string goes slack, the face spring (not shown) nudges the bowstring off the pulley, and the leg extends to full length. Not shown are the servo body or the leg. The winding direction and string displacement alternate left-right. Reprinted from Brown and Zeglin (1998) ©[1998 IEEE].

110
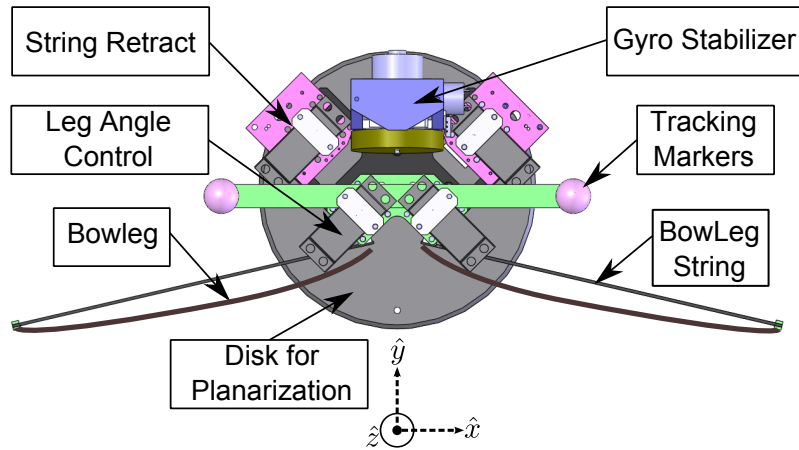
Figure 5.3: CAD design of the ParkourBot mechanism.

miniature version of this, but reliable release of the string could not be achieved at the smaller scale. A new mechanism was conceived and built, as shown in Figure 5.4. In the current design, a telescoping slider (green) engages the driven arm (orange) when the string is under tension, allowing the arms to move together, pull the string and retract the leg. When the string goes slack due to foot contact with the wall, the slider retracts and disengages from the drive arm, releasing the stored energy when the foot lifts off the wall.

5.1.4 Gyro-stabilizer

To maintain a constant orientation of the main body of the robot in the $\hat{z}$ direction, we have implemented a gyro-stabilizer. The gyro-stabilizer is a fast spinning flywheel mounted on a single-axis gimbal. The gyro spin axis is nominally aligned with the air table surface and the "vertical" axis ($\hat{y}$) of the climber. The gimbal permits the gyro to precess about the $\hat{x}$-axis, stabilizing the attitude about the $\hat{z}$-axis. A motor and potentiometer attached to the gimbal axis allow active correction of the body orientation. In our experiments, where the goal is to keep the body orientation constant, such corrections are rarely needed.
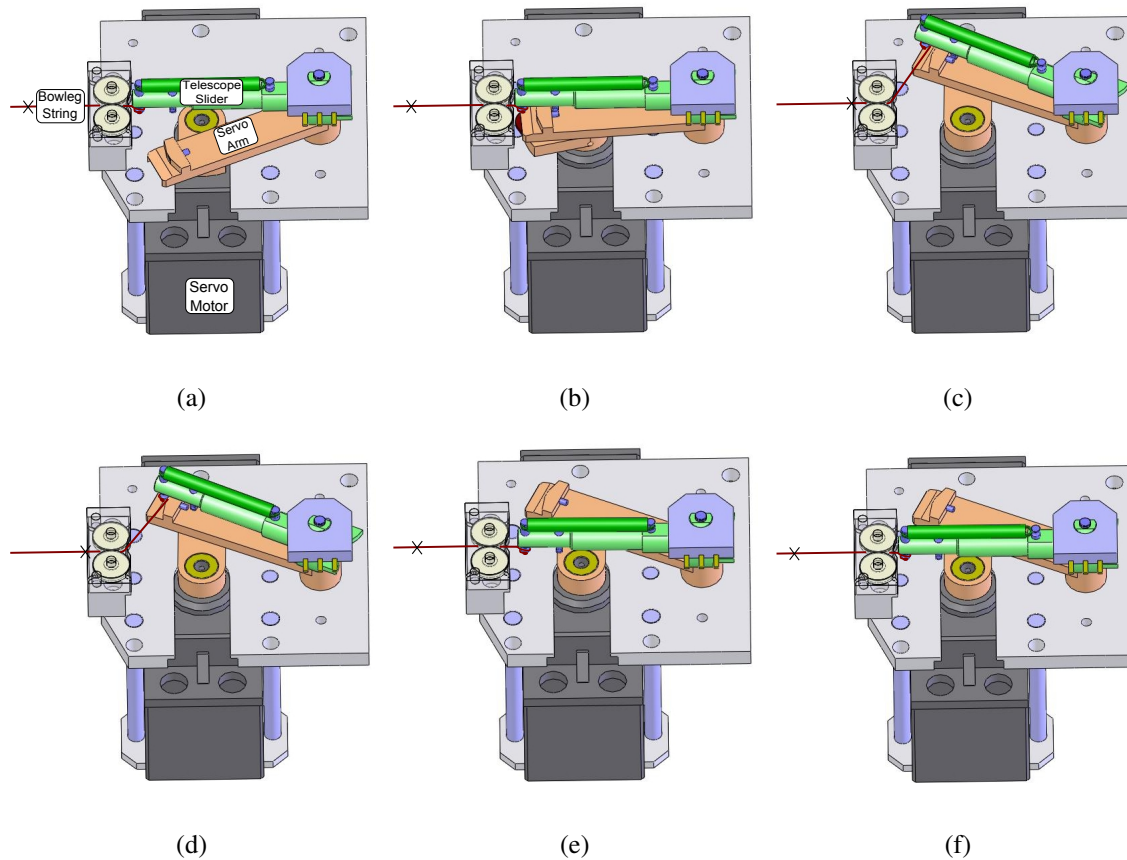
Figure 5.4: Retract mechanism sequence. The string is marked with an 'x' for length reference. (a) Leg extended, telescope slider fully extended. (b) Servo arm rotates and engages (couples) with latch pin. (c) Servo arm and telescope slider rotate together to retract leg. (d) After wall impact, string goes slack and the spring pulls the telescope slider back. This causes the latch pin to disengage. (e) The telescope slider is pulled back to center by tension in the bow string. (f) While leg extends and tension builds back in the string, the telescope slider extends and the sequence repeats.
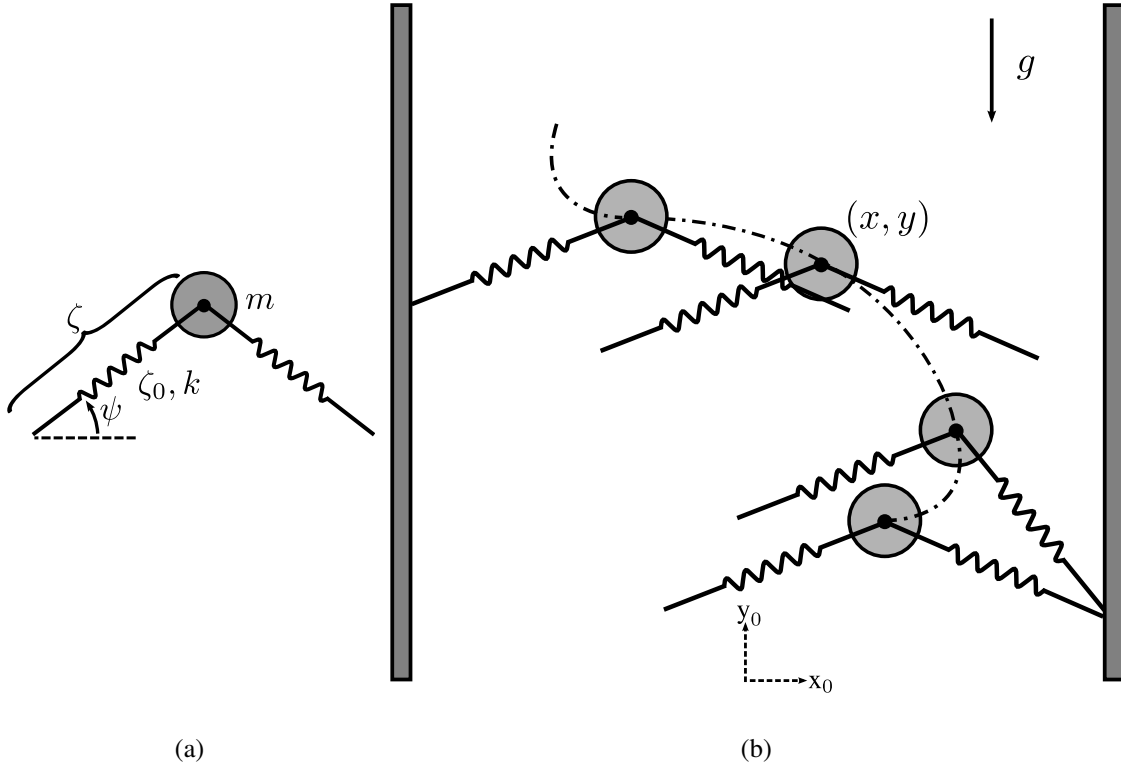
Figure 5.5: Mechanism schematics and configuration variables: (a) configuration variables during stance phase; (b) one full sequence including touch down on right wall, stance phase and flight phase toward left wall.

5.2 Modeling

A two-legged SLIP model is used for our mechanism. Two massless legs with rest length l_0 and stiffness k are attached to a point mass m as in Figure 5.5. Leg angle ψ_0 and leg length ζ_0 at touch down are the controls of the system, where $\zeta_0 \leq l_0$ determines the energy stored in the leg. Two parallel walls are at distance d apart, the gravitational acceleration is g and we assume no slip at contact. The inertial frame is centered between the walls. An entire stride of the SLIP model is composed of flight phase, touch down, stance phase, lift off and back to flight phase. Despite the simplicity of the model, during stance phase the system is a two degree-of-freedom Hamiltonian system without an explicit solution. Thus analysis and simulation in the later sections are done numerically in Matlab$^{\text{TM}}$.

During flight phase, the motion is ballistic and integrable, with configuration variables (x, y). During stance phase it is convenient to replace the cartesian coordinates with polar coordinates and represent the configuration as (ζ, ψ). Since the leg is massless, touch down occurs without impact when the end of the leg touches the ground. Lift off occurs when the leg length ζ reaches the spring resting length, l_0, and $\dot{\zeta}$ is positive. After liftoff the leg angle repositions back to ψ_0.

To better capture the physical system, we add two damping terms to the stance phase equations of motion, c_ζ and c_ψ, acting along the leg and at the pivot with the wall.

The Lagrangian in polar coordinates during stance phase is

$$L = \frac{1}{2}m(\dot{\zeta}^2 + \zeta^2\dot{\psi}^2) - \frac{k}{2}(l_0 - \zeta)^2 - mg\zeta\sin(\psi). \tag{5.1}$$

To find the nondimensional equations, we rescale with characteristic length l_0 and characteristic time $\sqrt{\frac{l_0}{g}}$. This converts the system into nondimensional variables $\hat{\zeta} = \frac{\zeta}{l_0}$, $\hat{\psi} = \psi$ and $\hat{t} = \sqrt{\frac{g}{l_0}}t$. The nondimensional Lagrangian becomes

$$\hat{L} = \frac{1}{2}mgl_0(\dot{\hat{\zeta}}^2 + \hat{\zeta}^2\dot{\hat{\psi}}^2) - \frac{k}{2}l_0{}^2(1 - \hat{\zeta})^2 - mgl_0\hat{\zeta}\sin(\hat{\psi}). \tag{5.2}$$

After adding viscous damping for both linear and rotational motion during stance phase we arrive at a set of nondimensional equations of motion

$$\ddot{\hat{\zeta}} = \alpha - \alpha\hat{\zeta} + \hat{\zeta}\dot{\hat{\psi}}^2 - \sin(\hat{\psi}) - \hat{c}_\zeta\dot{\hat{\zeta}}$$
$$\ddot{\hat{\psi}} = -\frac{1}{\hat{\zeta}}(\cos(\hat{\psi}) - 2\dot{\hat{\zeta}}\dot{\hat{\psi}}) - \hat{c}_\psi\dot{\hat{\psi}}, \tag{5.3}$$

where $\alpha = \frac{kl_0}{mg}$ is the ratio between the maximum spring force and the gravitational force. The ratio α gets larger as the spring gets stiffer or gravity shrinks.

5.2.1 Analysis and simulation

For the analysis section and the experimental section we use the parameters in Table 5.1. The BowLeg spring coefficient, k, was empirically approximated by measuring the displacement to a known load. Damping coefficients, c_ζ and c_ψ, were approximated from experimental data and drop tests.

Table 5.1: ParkourBot parameters for analysis and experiment sections

Dimensional Parameters		
Parameter	Description	Value
m	body mass	1.54 kg
l_0	leg rest length	0.223 m
d	wall width	0.54 m
k	leg spring stiffness	525 $\frac{\text{N}}{\text{m}}$
g	gravitational acceleration	0.98 $\frac{\text{m}}{\text{s}^2}$
c_ζ	damping coef.	1 $\frac{\text{Ns}}{\text{m}}$
c_ψ	damping coef.	1 Ns
Nondimensional Parameters		
α	nondimensional (ND) force - $\frac{kl_0}{mg}$	77.57
$\hat{d}$	ND wall width - $\frac{d}{\zeta_0}$	2.42
$\hat{c}_\zeta$	Damping coef. - $\frac{c_\zeta}{\sqrt{g/l_0}}$	0.4770
$\hat{c}_\psi$	Damping coef. - $\frac{c_\psi}{g/l_0}$	0.2276

5.2.2 Poincaré map and Poincaré section

The bipedal SLIP model is a hybrid system characterized by separate continuous flows (flight / stance phase) connected by discrete transitions (touch down / lift off). We use a Poincaré map to convert the hybrid system into a discrete-time system. Using the symmetry of the system we "flip" coordinates during touch down at the right wall and define the Poincaré map as a jump from the left wall to the right wall back to stance phase at the left wall. We choose the Poincaré section at touch down after the coordinate "flip". The touch down position x is calculated from ζ_0 and ψ_0. The state variable y does not appear in the Poincaré section. We do not want to stabilize it, and due to the vertical symmetry it does not appear in the equations of motion for the other state variables. We are left with only two state variables, $\dot{x}$ and $\dot{y}$, in our Poincaré section.

5.2.3 Fixed-point and local stability

A period-n fixed-point is a point on the Poincaré section that is mapped back to itself after applying the Poincaré map $\mathbf{P}$ n times. To find a period-n fixed-point q^*, with the controls ζ_0 and ψ_0 fixed, we use the Levenberg-Marquardt gradient descent method (Gill et al. (1981)) to find the zeros of $\mathbf{P}^n(q) - q$, where n is the periodicity of the desired fixed point, and q is the state on the Poincaré section. The gradient is calculated numerically.

To find the local stability of an orbit, we linearize around the fixed-point by computing a forward difference approximation to the Jacobian. A fixed-point is stable if and only if both of its eigenvalues are inside the unit circle in the complex plane.

In experiments, we have noticed an asymmetry of the two leg controls due to imperfection of the servo motor controller. To simulate this asymmetry, we introduce alternating controls (ψ_0 and ζ_0) between Poincaré maps. On odd jumps, ψ_{0L}, ζ_{0L} are used, while on even jumps, ψ_{0R}, ζ_{0R} are used. For the symmetric case with identical controls, we searched for stable period-1 motions. Depending on the fixed controls and the initial conditions of the search, we found representative stable period-1 gaits for descending, jumping in place, and climbing upwards. For

the asymmetric case, we found stable period-2 and period-4 gaits for climbing, jumping in place, and descending. We did not search exhaustively for all stable gaits over all possible controls, and the solutions shown here are local attractors for particular gradient searches we tried. Figure 5.6 depicts a stable period-1 gait for the symmetric case, and stable period-2 and period-4 gaits for the asymmetric case.

5.2.4 Varying energy input

As Figure 5.7 shows, there is a high correlation between energy input (the amount of leg retraction) and the stability of the system. Higher input energy corresponds to higher local stability and a larger basin of attraction. We vary the energy input by changing the amount of leg retraction, then compute the basin of attraction and fixed-point for that particular leg retraction.

5.3 Experiments

5.3.1 Experimental setup

As in the open-loop DSAC experiments, we use a tilted air table (see Figure 3.21) as the testbed, which enforces the planar constraint on the ParkourBot, and provides an easy way to change the effective gravity by tilting the table. Friction between the rubber foot and wall was observed to be about 2.0, thus making foot slipping rare. A $NaturalPoint^{TM}$ FLEX V100 IR camera is mounted above the air table. The camera provides positions of all IR reflectors to a PC, which communicates with the robot over wireless XBee protocol.

5.3.2 Open-loop experimental results

The goal of the open-loop experiments was to verify our analysis results, particularly the existence of period-1, period-2, and period-4 stable cycles, and the correlation of stability with input energy. We readily observed stable period-2 and period-4 motions (Figure 5.9), but not period-1

117
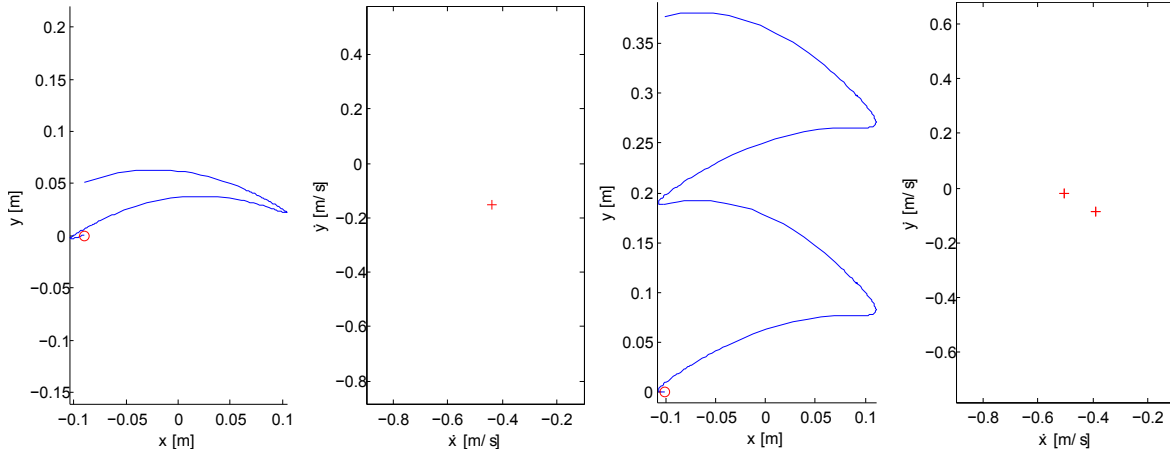
(a) stable period-1 with symmetric controls
(b) stable period-2 with asymmetric controls
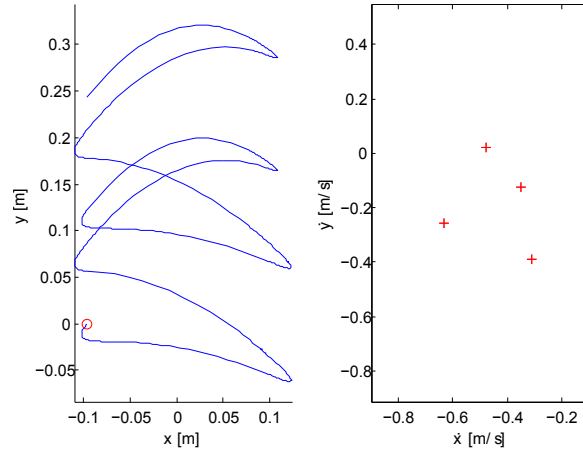
(c) stable period-4 with asymmetric controls

Figure 5.6: Trajectories and fixed-points of stable gaits. Left column shows the trajectory of two cycles starting from a fixed-point (red circle), and right column shows the fixed-points on the Poincaré section (red '+'). The magnitudes of the maximum eigenvalues are 0.9007, 0.8652 and 0.5746 respectively. Controls: (a) $\psi_{0L} = \psi_{0R} = 30°$ and $\hat{\zeta}_{0L} = \hat{\zeta}_{0R} = 0.93$. (b) $\psi_{0L} = \psi_{0R} = 30°$, $\hat{\zeta}_{0L} = 0.87$ and $\hat{\zeta}_{0R} = 0.9$. (c) $\psi_{0L} = 30°$, $\psi_{0R} = 35°$, $\hat{\zeta}_{0L} = 0.9$ and $\hat{\zeta}_{0R} = 0.95$.
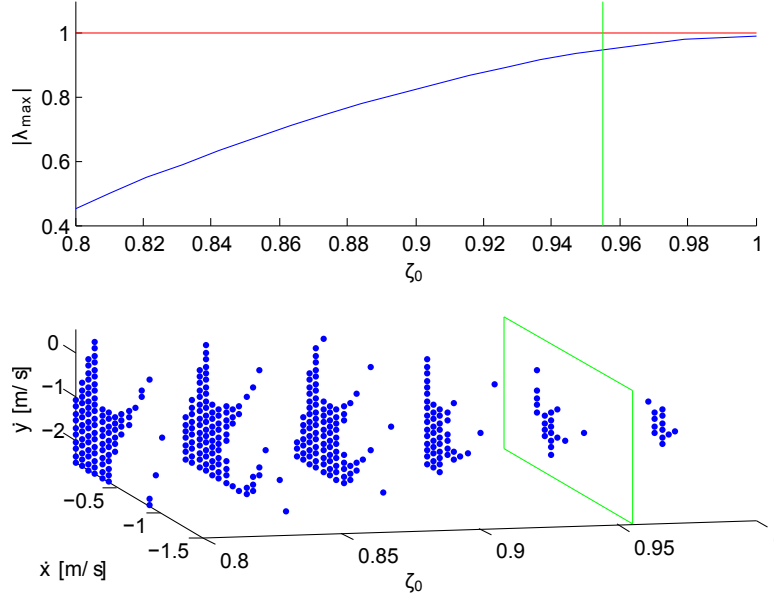
Figure 5.7: Varying $\hat{\zeta}_0$ with symmetric controls. All fixed-points are period-1 motions. Green line / plane indicates critical $\hat{\zeta}_0$ for jumping in place, leg lengths to the right of green line / plane correspond to climbing down. Upper graph shows the maximum magnitude of eigenvalue of the Jacobian for the fixed-point given $\hat{\zeta}_0$. Lower graph shows the basin of attraction for various energy levels.

motions, perhaps due to the asymmetric leg angles noted earlier. To measure stability we use Mean Jumps To Failure (MJTF), similar to Byl and Tedrake (2009). We chose ten different energy levels and ran eight experiments for each energy level. For each individual experiment, we counted the number of jumps before crash, as seen in Figure 5.10. This plot correlates to our simulation results in Figure 5.7, showing how the system is more stable when the input energy is increased. Typically the main failure mode is the body orientation drifting to the point where the state of the robot exits the basin of attraction.

5.3.3 Closed-loop experimental results

The goal of the closed-loop experiments was to assess the feasibility of stabilizing the vertical height of the robot. We determine height error from IR tracker data at the Poincaré section (at touch down), and use a PID controller to determine energy input for the next touch down. Figure 5.12 shows the successful stabilization at a height of 55 cm. See supplementary attachment for a video of this experiment.

5.3.4 Discussion

While there is qualitative agreement between simulation, analysis, and experiment, there are also differences arising from modeling approximations and the limitations of our experimental apparatus. While we model the robot body as a point mass, the actual robot body has pitch which drifts, ultimately leading to failure when leg angles reach joint limits. We have already noted the asymmetry in the robot mechanism, which may be responsible for the lack of observed period-1 stable gaits. We also lack an accurate launching mechanism. Finally, since the most stable gaits involve the most rapid climbing, our most successful experiments are limited by the height of tilted table.
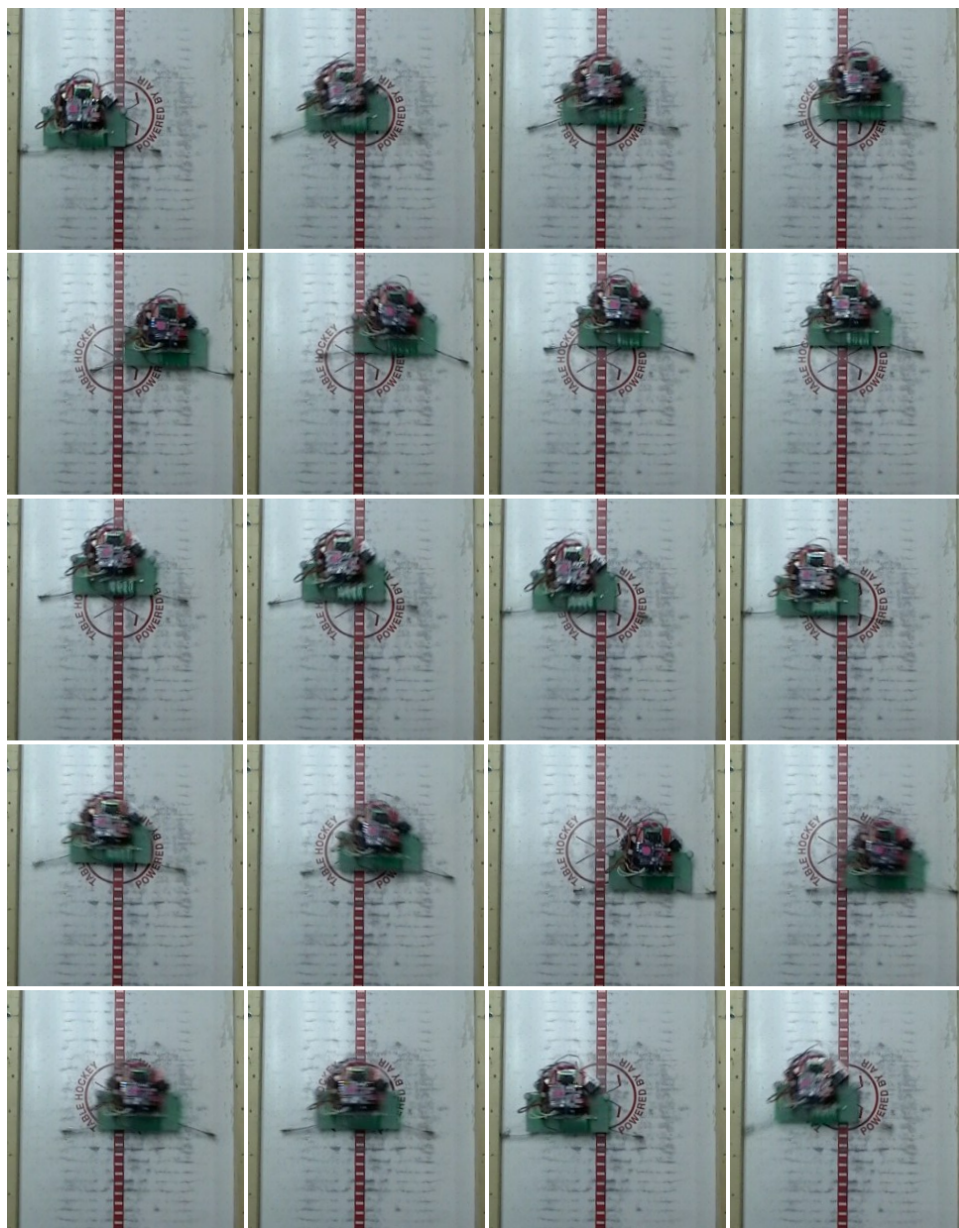
Figure 5.8: Period-4 experiment.
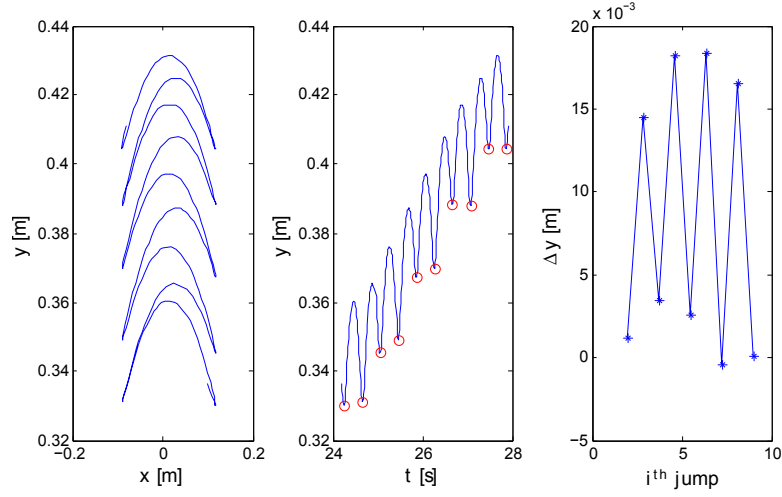
(a) Period-2
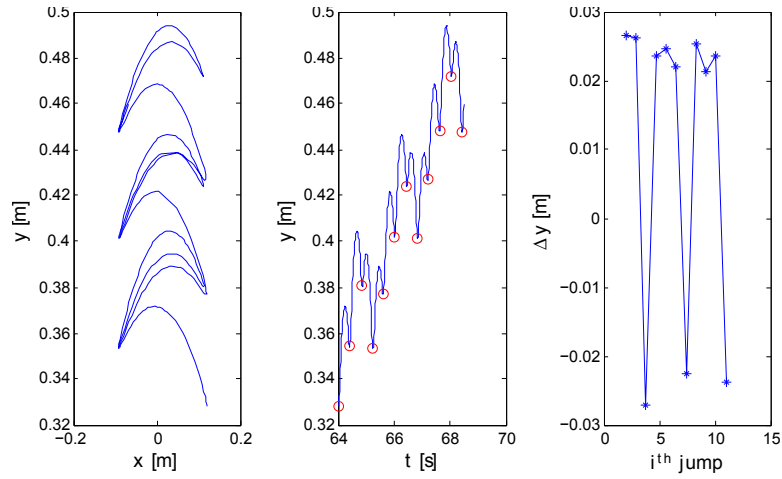
(b) Period-4

Figure 5.9: Experimental data of open-loop climbing: climbing trajectory (left column), height y vs. time (middle column), where the red circle indicates the Poincaré section, and Δy between pairs of consecutive states on Poincaré section (right column). (a) Stable period-2 climbing up. (b) Stable period-4 climbing up.
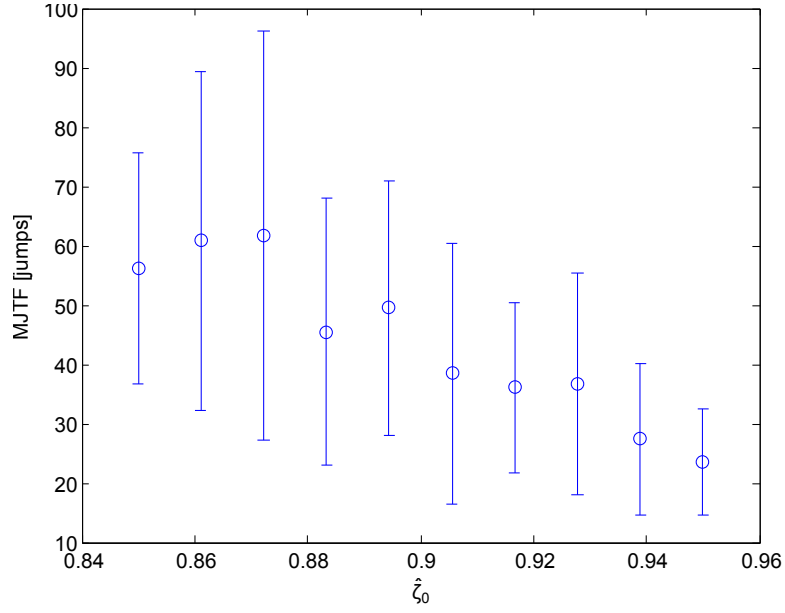
Figure 5.10: Mean Jumps To Failure (MJTF) vs. leg retraction $\hat{\zeta}_0$. Vertical axis shows the average number of jumps before crash, with error bars representing the standard deviation. Leg retraction $\hat{\zeta}_0$ was roughly approximated. Eight experiments are conducted for each energy level.
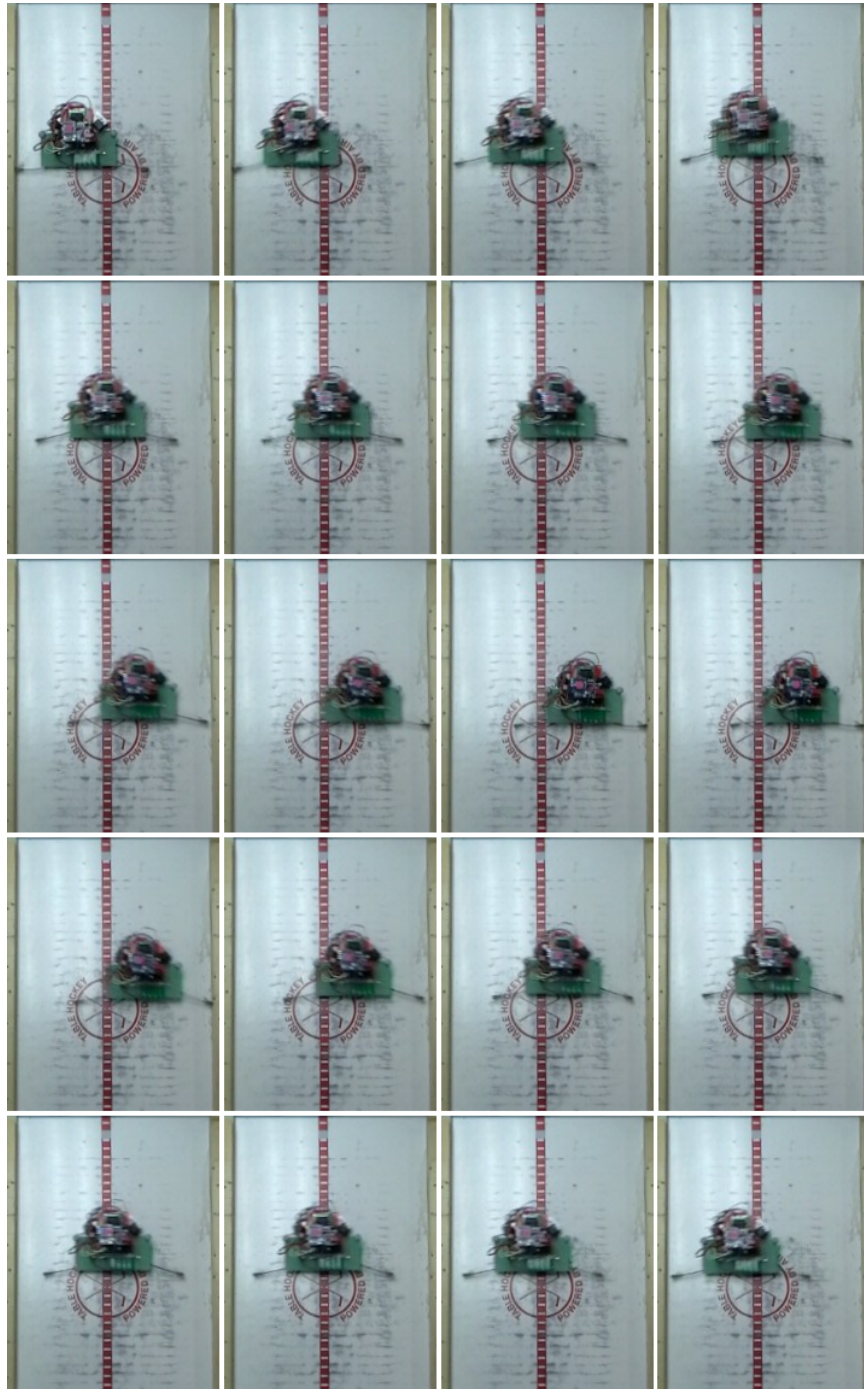
Figure 5.11: Maintaining constant height experiment
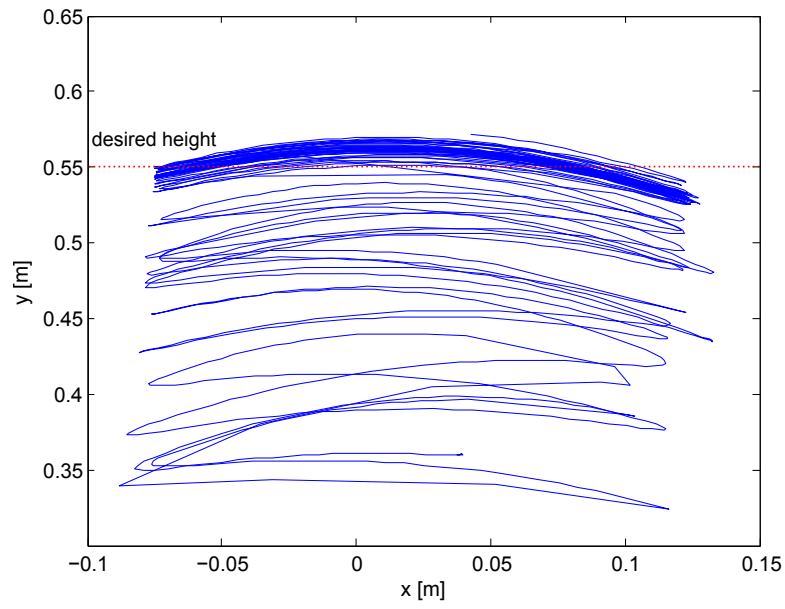
(a) Climbing trajectory
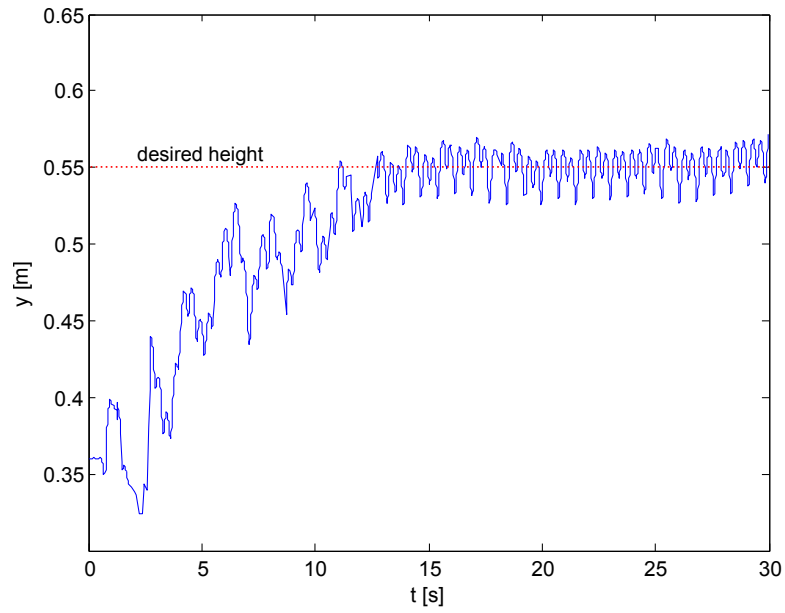
(b) Height vs. time

Figure 5.12: Experimental data of closed-loop climbing reaching a desired height of 55 cm: (a) climbing trajectory; (b) height y vs. time.

Part II

Closed-Loop Climbing

Chapter 6

Closed-loop algorithm

So far, we have shown how the use of the natural stability characteristics of the DSAC, Parkour-Bot and the DTAR can be used to climb in a simple environment without the use of sensors, i.e., open-loop climbing. A locally stable system is stable even with small changes in environment or mechanism parameters. However, for a larger change in the environment the mechanism will not stay stable in general and a change in controls must be used to keep the mechanism stable. The "classic" closed-loop control in robotics uses sensors to estimate the state of the system to predict upcoming changes in environment. As an example, the ParkourBot control could consist of estimating during flight phase where the trajectory is heading, together with estimating the shape of this foothold. The control loop will then use these inputs to modify the leg angle controls. The higher level planner will usually also take into account what the outgoing trajectory should be in order to plan ahead for the next foothold. This control methodology requires the ability to sense the current state fast and accurately. The control scheme proposed here is minimalistic in nature. We seek to find the simplest complexity increase in sensing, planning and control for a small increase in complexity of the environment.

This chapter introduces a "work in progress" method for controlling the mechanism described in previous chapters. Specifically, we will introduce a discrete change in terrain, wall width, at

specific times. In order to enable this kind of terrain transition, a control will be changed to enable a safe transition of terrains. This method will use the knowledge gained in previous chapters, regarding open-loop stability, to find a valid transition of controls. The control algorithm, assuming knowledge of the upcoming environment transitions, plans controller switches that stabilize climbing across transitions.

The approach of this method is to compose a transition graph between different controls and different terrains. Each node in the graph is a set of controls and terrain (environment) and directed edges are valid (stable) transitions between these nodes. A valid transition can only include a change in controls or a change in terrain but not both. For instance, the ParkourBot will change leg angle in the first terrain, followed by a change in wall width with a fixed leg angle. In order to safely switch between controllers, the attractor of one controller needs to be enclosed inside the basin of attraction of the next controller. Both the attractor and the basins in our systems are on the Poincaré section and not on the full state space. Similar to the sequential composition work of Burridge et al. (1999) we find a sequence of controls to safely transition between two environments. This chapter begins with an illustrative example using the SLIP biped model. We then formalize the algorithm and implement and simulate it on the DSAC and ParkourBot mechanisms. We conclude this chapter with experimental results of the ParkourBot mechanism.

6.1   Illustrative example – bipedal running

To illustrate the proposed method we start with a simple example of a biped hopper traversing a step-down. We use a simplified model of the biped hopper, the Spring Loaded Inverted Pendulum (SLIP). We show how a biped hopper can switch between controls (leg angle) for the purpose of staying stable after it transitions to a new environment (new ground height).

The SLIP model is a massless spring attached to a point mass body as shown in Figure 6.1
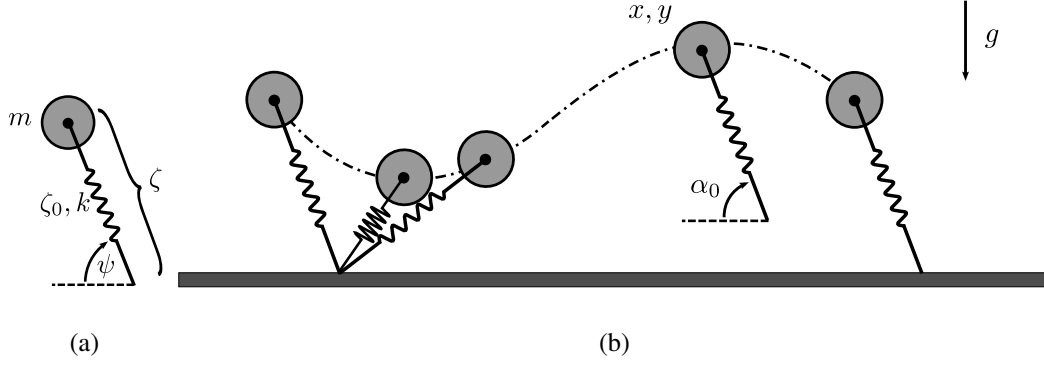
Figure 6.1: Slip model schematics.

constrained to move in the sagittal plane. The body is a point-mass connected through a frictionless pin-joint to the spring leg. During stance we assume that the infinitesimal point foot is connected to the ground through another frictionless pin-joint. An entire stride of the SLIP model is composed of flight phase, touch down, stance phase, lift off and back to flight phase. The SLIP model is a classical example of a hybrid system comprising a continuous flight phase which switches to a stance phase. The system's parameters are mass of the body (m), rest length of the spring leg (l_0), spring coefficient (k), gravitational acceleration (g), and landing leg angle (α_0). The systems state variables are $(x, y, \dot{x}, \dot{y}) \in \mathbb{R}^4$. During stance phase it is convenient to replace the cartesian coordinates with polar coordinates and represent the configuration as ζ and ψ as shown in Figure 6.1. The derivation of the equations of motion is similar to the ones found in (Seyfarth et al., 2002; Ghigliazza et al., 2005; Geyer et al., 2005). The kinetic energy of the system is

$$T = \frac{1}{2}m(\dot{\zeta}^2 + \zeta^2\dot{\psi}^2) \tag{6.1}$$

The potential energy of the system is

$$V = mg\zeta\sin\psi - \frac{k}{2}(\zeta_0 - \zeta)^2 \tag{6.2}$$

where we have picked the spring potential to be a conservative Hooke spring law, and ζ_0 is the resting length of the spring.

131

By using the Lagrangian formulation, the two equations of motion during stance phase are

$$\ddot{\zeta} = \zeta\dot{\psi}^2 + \frac{k}{m}(\zeta_0 - \zeta) - g\sin\psi \tag{6.3}$$

$$\zeta\ddot{\psi} = -g\cos\psi - 2\dot{\psi}\dot{\zeta} \tag{6.4}$$

Similar to the approach in (Seyfarth et al., 2002; Geyer et al., 2005), it is convenient to pick the Poincaré section at the apex during the flight phase, defined as

$$\Sigma = \{(x, y) \in \mathbb{R}^2 \mid \dot{y} = 0, \zeta\sin\psi > 0\} \tag{6.5}$$

Therefore, the Poincaré map is a one dimensional map from apex to next apex

$$\mathbf{P} : y_i \in \mathbb{R} \rightarrow y_{i+1} \in \mathbb{R}.$$

The example here will show how the SLIP hopping model with a leg angle α_1 can be stable in the first terrain but will be unstable after transitioning to the next terrain. By changing the leg angle to α_2, the mechanism is able to stabilize even after transitioning to the new terrain. A cartoon of this SLIP example is depicted in in Figure 6.2.

The main reason for using this model as an example is that the Poincaré map is one dimensional. For every set of mechanism and environment parameters (m, k, ζ_0, g) together with the control parameter (α_0) one can plot a 1-D return map of this system as in Figure 6.3. In general a return map gives a graphical representation of the Poincaré map from one Poincaré section to the next. This 1-D return map portrays a mapping from apex y_i to the next apex y_{i+1}. The intersection of the return map with the diagonal ($y_i = y_{i+1}$) yields the fixed points of the system. Moreover the slope at these fixed points distinguishes between locally stable and locally unstable fixed points. Similar to the investigation of local stability in terms of eigenvalues inside the unit circle in Section 2.2, any slope around the fixed point between $-45°$ and $45°$ is stable. A slope of $45°$ is marginally stable, and above $45°$ it is unstable. See Strogatz (1994) for more on this
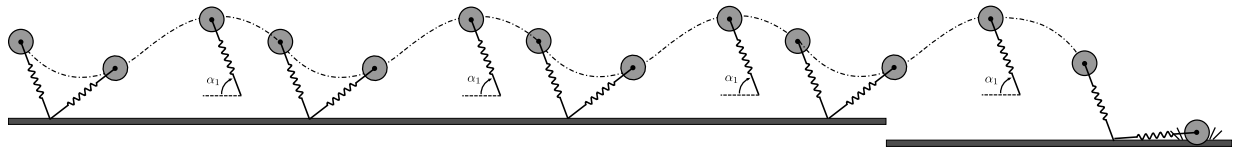
(a) Failed transition
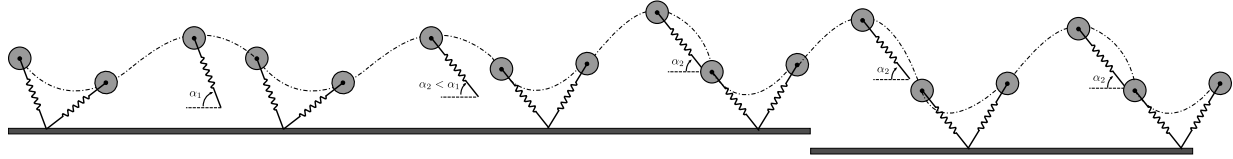
(b) Successful transition

Figure 6.2: SLIP step down transition. (a) Failing to traverse the step down without switching controllers. (b) A successful step down by switching controllers before the step.
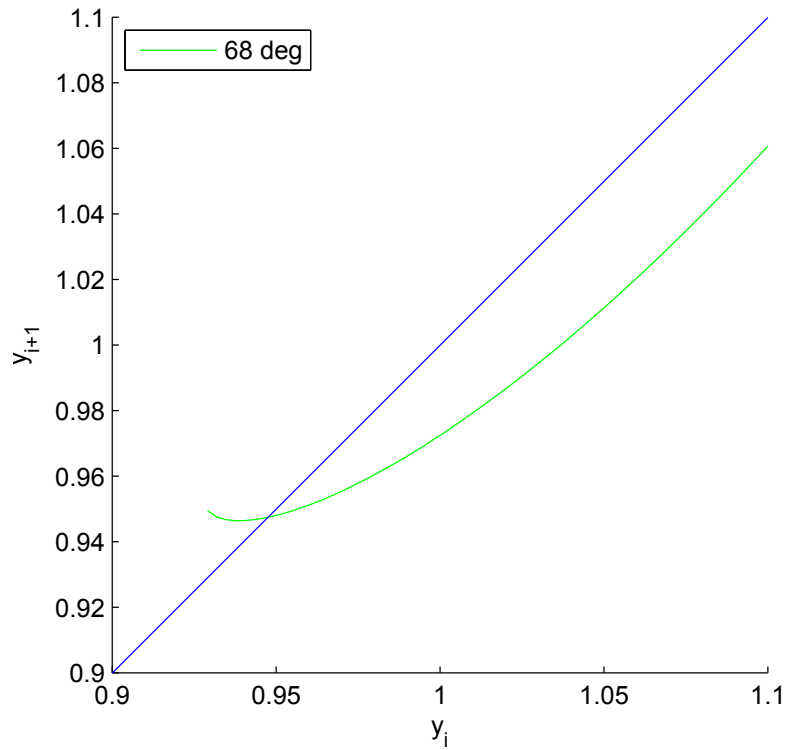
Figure 6.3: 1-D Return map for leg angle $\alpha_0 = 68°$. The fixed point of the system is the intersection of the green line, representing the return map, with the diagonal line. The slope of the return map at the fixed point is less than $45°$, hence it is stable.

subject. We use similar human parameters as where chosen in Geyer et al. (2002), specifically, $\zeta_0 = 1$m, $m = 80$kg, g=0.98 $\frac{\text{m}}{\text{s}^2}$, $k = 20\frac{\text{kN}}{\text{m}}$, and initial horizontal velocity of $\dot{x} = 4.7\frac{\text{m}}{\text{s}}$. Using these parameters and plotting the 1-D return maps for eight different leg angles (α_0), we obtain the plots in Figure 6.4. Figure 6.4(a) was plotted for a nominal ground terrain. For a different terrain with a 2cm drop the eight different return plots are plotted on top of the original return maps in Figure 6.4(b). The advantage of using such a simple system with a 1-D return map, is that we can now observe and find possible controlled transitions (in the same environment) and the passive switches between a fixed point in the first environment to the next environment staying with the same parameters. Figure 6.5 portrays an unsuccessful transition between 0cm level to -2cm, which results in a failure. Starting at level ground (0cm) with a leg angle of $69.1429°$ (solid orange line) 6.5(a) will converge to the fixed point 6.5(b). After transitioning to the next terrain (dashed orange line) 6.5(c) the next return map will be outside the feasible range and will fail 6.5(d).

Figure 6.6 depicts an effective transition. In order to obtain this transition, the controls (leg angle) must be changed before the terrain changes. Figure 6.6(a) begins at the fixed point with $69.1429°$ leg angle. Before arriving to the new terrain the mechanism changes to a smaller leg angle of $67.4286°$ (Figure 6.6(b)). After converging to the fixed point Figure 6.6(c) the passive switch occurs, i.e., the switch to the new -2cm terrain. As can be seen in Figure 6.6(d) the new leg angle controls is able to converge to the new fixed point in the new terrain.
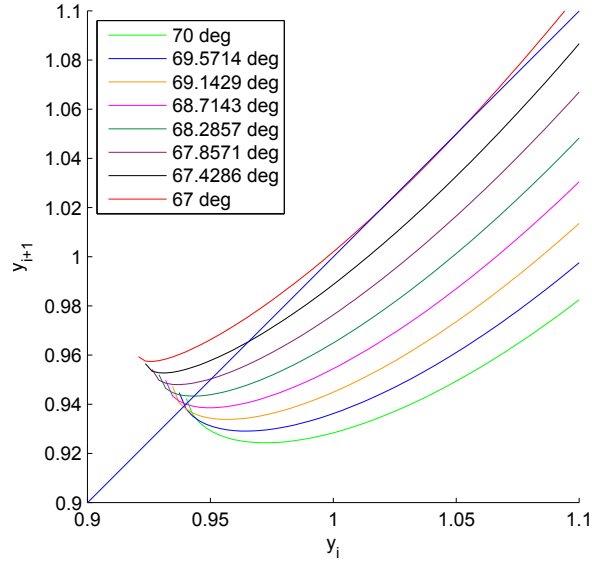
(a) 1-D return maps for ground level = 0cm. Fixed point of $67°$ is marginally stable. Fixed points of $67.4286° - 69.1429°$ are stable. Fixed points of $69.5714°$, and $70°$ are not stable.
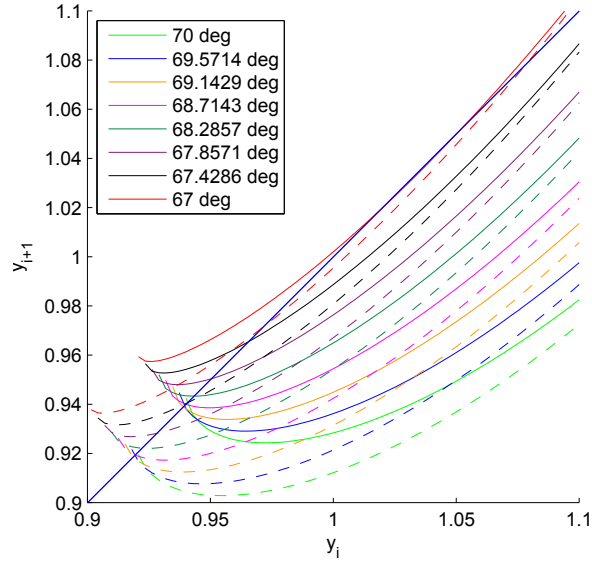
(b) 1-D return maps for ground level = 0cm (solid lines), and -2cm (dashed lines).

Figure 6.4: 1-D Return map for eight different leg angles (α_0) for (a) ground height of 0cm, and (b) overlayed on top of -2cm ground (drawn as dashed lines).
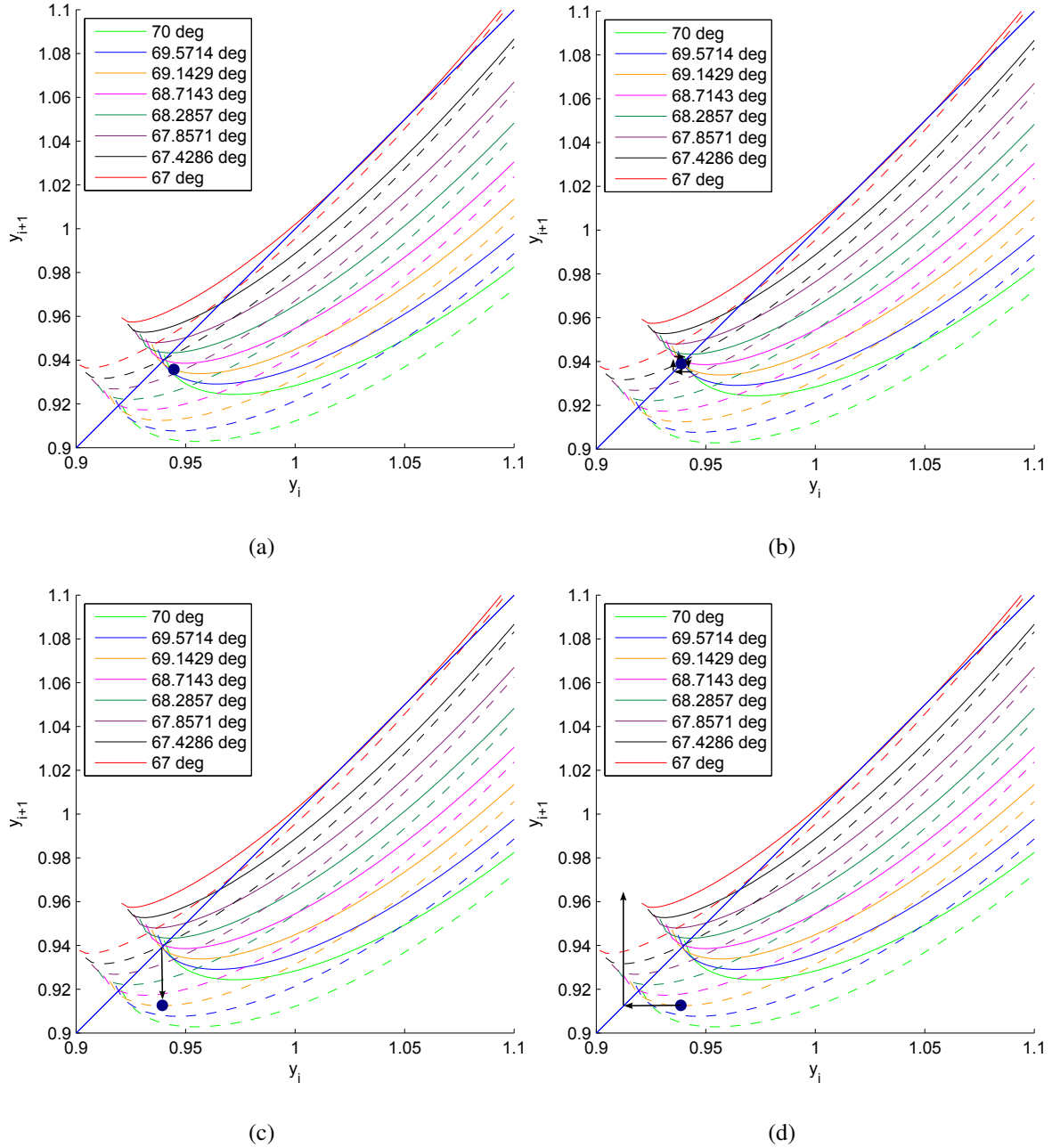
Figure 6.5: Unsuccessful transition from 0cm level to -2cm. Starting at level ground (0cm) with a leg angle of $\alpha_0 = 69.1429°$ (solid orange line) (a) will converge to the fixed point (b). After transitioning to the next terrain (dashed orange line) (c) the next return map will be outside the feasible range and will fail (d).

136
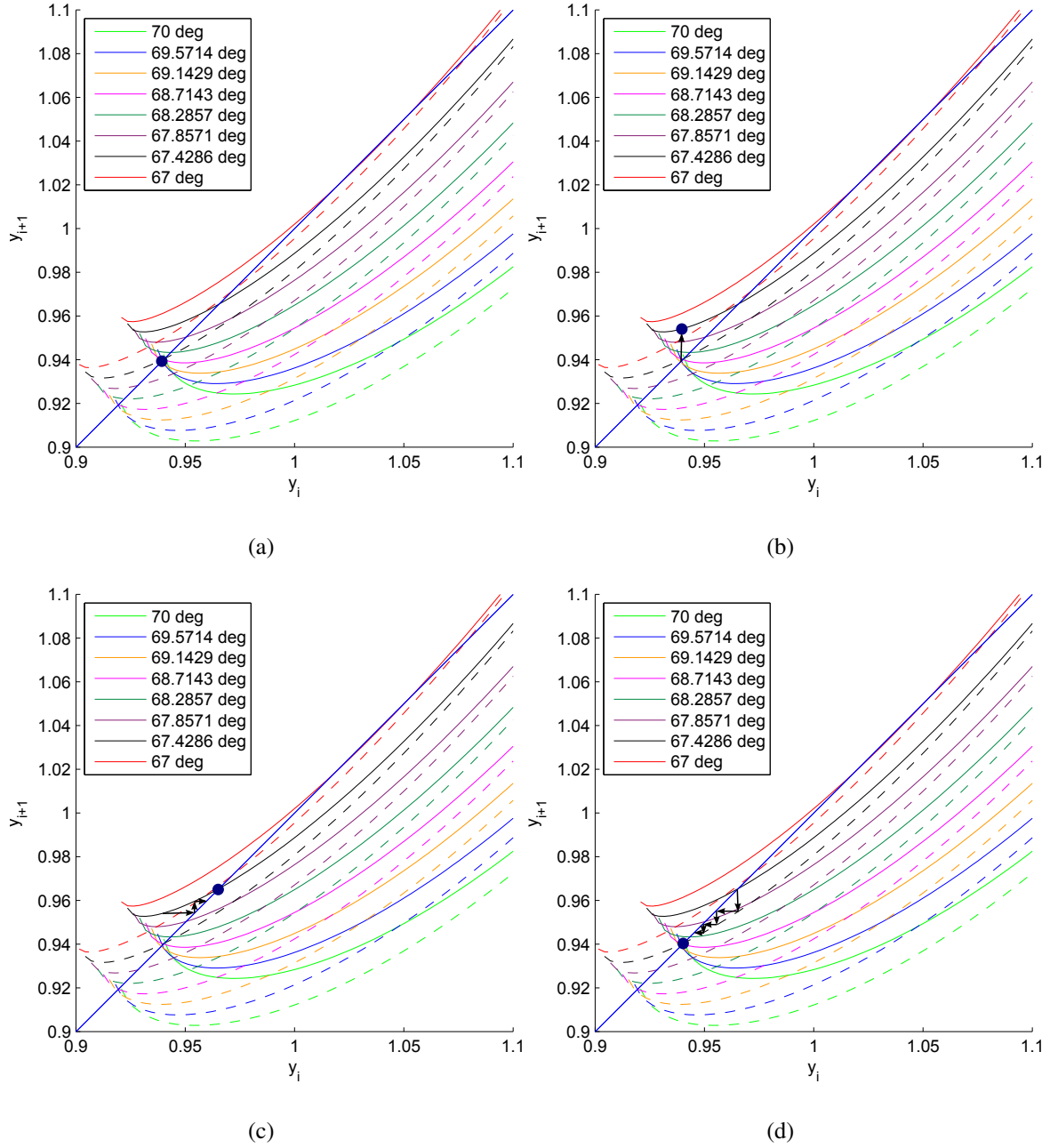
Figure 6.6: Successful transition. The mechanism begins at the fixed point with $\alpha_0 = 69.1429°$ leg angle (a). Before arriving to the new terrain the mechanism changes to a smaller leg angle of $\alpha_0 = 67.4286°$ (b). After converging to the fixed point (c) the passive switch occurs, i.e., the switch to the new -2cm terrain. As can be seen in (d) the new leg angle controls is able to converge to the new fixed point in the new terrain.

137

6.2   Algorithm and implementation

As exemplified in the previous section on the SLIP model, the motivation of this control scheme is that given a known map of the environment we must come up with preplanned transitions of controllers that will keep the mechanism stable across environment transitions. The next section implements this algorithm on the DSAC and ParkourBot mechanisms. We assume that the environment is known a priori. Moreover, changes in environment are only executed after the mechanism is stabilized into its attractor. This is a rather strong assumption which might later be relaxed by including the time taken to stabilize to the area around the attractor and using this information in the planning phase. Finally, we assume that the environment change, which in this case is represented by a change in wall width, will occur independently of the vertical location of the mechanisms. That is, even if the climber climbed downwards but has stabilized into its attractor, the next wall width change will be executed.

Two kinds of transitions exist. The first is the change in controls while remaining in the same environment, called *controlled switch*. The second kind is where the controls are fixed but the environment switches, called a *passive switch*. In general a passive switch occurs when the robot reaches the new environment, however we assume that this switch will only occur after the robot reaches the attractor of the current controller.

In general, when a passive switch occurs and the mechanism changes terrain, the stable attractor of the first terrain will be mapped to a new location in the new terrain. This can be seen in the animation in Figure 6.7 where the DSAC transitions through a change in wall width and the steady state angle θ_1 changes to θ_2 at the new environment. This fact will be used in the algorithm described below, when calculating if one attractor in one terrain is inside the basin in another terrain.

An illustration of the proposed algorithm is given in Figure 6.8. Here we portray the basins of attraction as funnels. These funnels represent the flow of the dynamic system from the initial
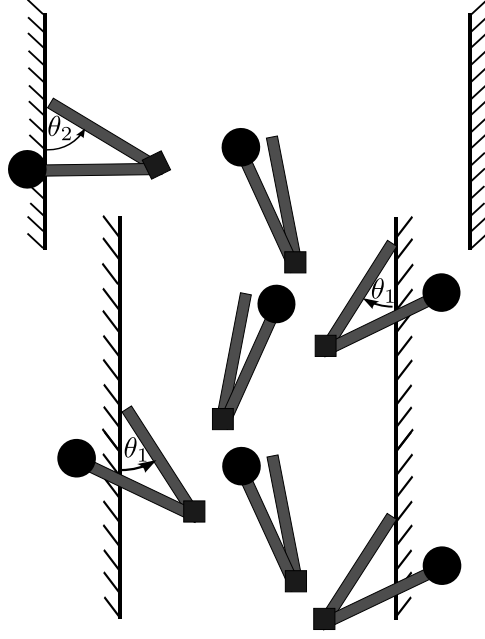
Figure 6.7: Changing wall width. Fixed point in bottom wall width (θ_1) changes in new terrain (θ_2).

state to the attractor. The green cross sections on the funnels represent the Poincaré sections. The goal of this algorithm is to allow controlled transitions in one environment to the point where a passive switch of the environment won't perturb the stability of the system.

The outline of the algorithm is given in Algorithm 1. We first construct the adjacency matrix which converts into a transition graph. This adjacency matrix, as explained in Algorithm 1, is a matrix which contains 0 and 1. Elements with 1 represent an adjacency which in turn represent a valid transition between nodes in the transition graph. In this algorithm we assume that only one period-1 attractor exist per terrain and controller. A few notes about the implementation of this algorithm. The algorithm requires an approximation of the basin of attraction of each attractor, for which several methods can be used. We use the cell mapping technique as described in Chapter 3. Since, in most cases we are interested in a set of points around the attractor, we implemented a simple erosion algorithm to shrink the basins of attraction by a constant radius.
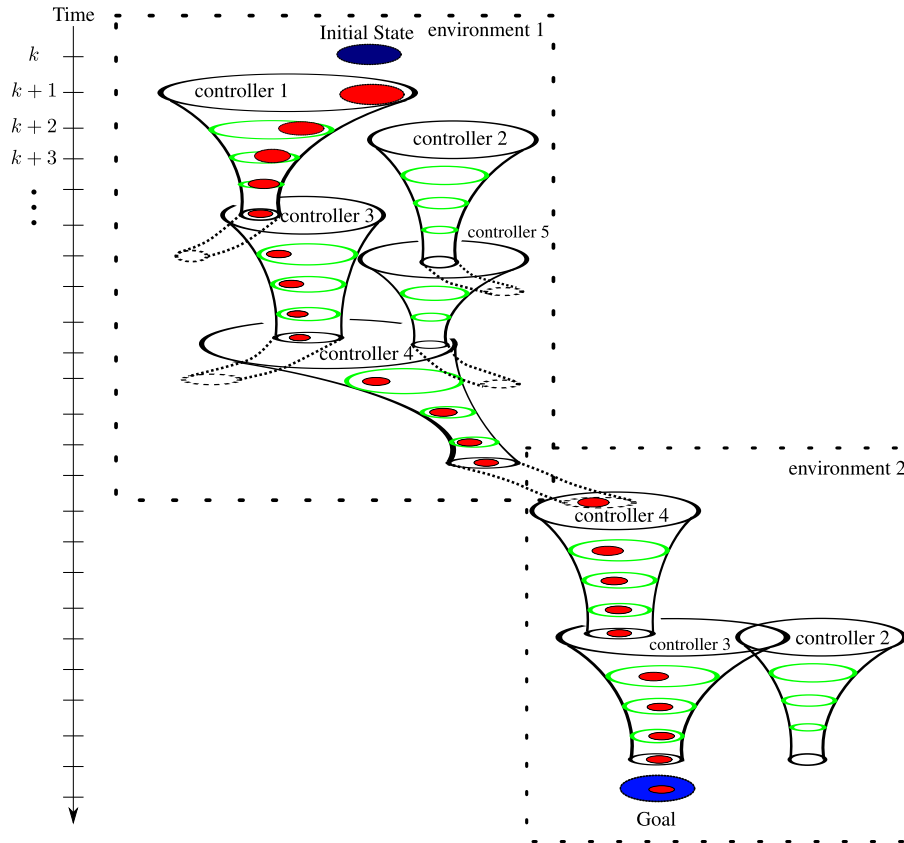
Figure 6.8: Schematics of five different controls in two different terrains. The controls might be different amplitude and frequencies and the terrain is the wall width in the case of the DSAC. The task is to find a plan from initial state (purple ellipse) to goal (blue ellipse) by switching controllers. The goal is at the second environment (terrain). The funnels represent the flow of the dynamic system. The mouth of the funnel (the inlet) represents the basin of attraction on the Poincaré section. The green cross sections represent the discrete Poincaré mapping. The dashed funnels represents a passive switch to the new environment, as explained in Figure 6.7. Only three controllers are drawn in environment 2 – the other two might be non valid controllers with empty basins of attraction. A viable path (marked with red ellipses) from start to goal, is sequencing in environment 1 through controllers 1, 3, 4, then passively switching to environment 2 with the same controller 4, and finishing with a switch to controller 3.

140

Algorithm 1 Building the adjacency matrix and transition graph

- Discretize controls space and environment space. Let $\{\mathcal{T}_i\}i = 1, 2, ...t_{num}$ be the different terrain. Let $\{\mathcal{C}_j\}j = 1, 2, ...c_{num}$ be the different controllers.

- Approximate the basin of attraction for each controls and environment. The attractor cells are recorded as $attractor_{i,j}$ and the basin cell as $basin_{i,j}$.

- Let $A = (a_{i,j})$ be an $n \times n$ adjacency matrix corresponding to the transition graph, where $n = t_{num} \times c_{num}$.

- Since diagonal elements of the adjacency matrix correspond to transition from the vertex back to itself, all these entries will be assigned a 1.

- The adjacency matrix is split into block matrices. Each diagonal submatrix (of size $c_{num} \times c_{num}$) represents the transition in a fixed terrain. The top left submatrix $a_{i=1:c_{num}, j=1:c_{num}}$ is the adjacency matrix for the first terrain ($\mathcal{T}_1$), the second submatrix $a_{i=num+1:2c_{num}, j=c_{num}+1:2c_{num}}$ is the adjacency matrix for the second terrain ($\mathcal{T}_2$), etc.

- For each diagonal submatrix, representing one terrain, if an attractor of a controller is in the basin of another controller it will be assigned a 1. For example, in terrain $\mathcal{T}_1$, if an attractor of controller $\mathcal{C}_1$, is inside the basin of controller $\mathcal{C}_2$, $attractor_{1,1} \in basin_{2,1}$ then $a_{1,2} = 1$).

- The off diagonal submatrices correspond to transitioning between terrains. Since, while transitioning to a new terrain, we assume controller must be fixed, only the diagonal elements of these submatrices may be assigned a 1. These elements corresponding to transitioning between terrains are calculated by first mapping the attractor to the new terrain as was shown in Figure 6.7, and assigning 1 if the mapped attractor is in the basin of one of the terrains.

This achieves two tasks, first it disregards scattered or irregular cells that were recognized as part of the basin. Second, it allows to check if single point attractor is inside the basin which is similar to checking it a set of attractors are inside this basin.

The next two sections implement the algorithm on the DSAC and the ParkourBot mechanisms. Both will be exemplified with simulation. Experiments with the ParkourBot mechanism conclude this chapter.

6.3   DSAC implementation

The algorithm described above is implemented on the DSAC mechanism. Five different (half) wall widths were chosen: $d_{\text{wall}} = 0.03\text{m}, 0.04\text{m}, 0.05\text{m}, 0.06\text{m}, 0.07\text{m}$. Six sinusoid amplitudes $A = 0.5, 0.76, 1.02, 1.28, 1.54, 1.8\text{rad}$, were chosen as possible controllers, all with constant frequency $\omega = 8\frac{\text{rad}}{\text{s}}$. Leg inertia was arbitrarily chosen to be $I_1 = 1 \cdot 10^{-3}\,\text{kg}\,\text{m}^2$, the other mechanism parameters were kept as in Table 3.1 in Chapter 3. Using the cell mapping technique, the 30 basins of attractions were approximated. By implementing the algorithm described above, the adjacency matrix and the transition graph is obtained (shown in Figure 6.9).

Once the transition graph is obtained, it is possible to plan a sequence of controlled switches to reach from start to goal via way points. As an example, see Figure 6.10. Here we plan a path from the vertex with control $A = 0.5\text{rad}$ in $d_{\text{wall}} = 0.03\text{m}$ to the goal at $A = 1.8\text{rad}$ in $d_{\text{wall}} = 0.06\text{m}$ via $A = 1.02\text{rad}$ in $d_{\text{wall}} = 0.05\text{m}$. It may be the case that only the initial and final wall width are given, however we can make the problem more complex by appointing way points with specific controls. These example were planned by hand, but a graph search algorithm such as A* or Dijkstra's should be used to find a path.

After finding the shortest path shown in Figure 6.10 we simulated this path using the model from Chapter 3. Figure 6.11 shows the simulated path.

As seen in the simulated motion in Figure 6.11, in the initial terrain and controls, the mecha-
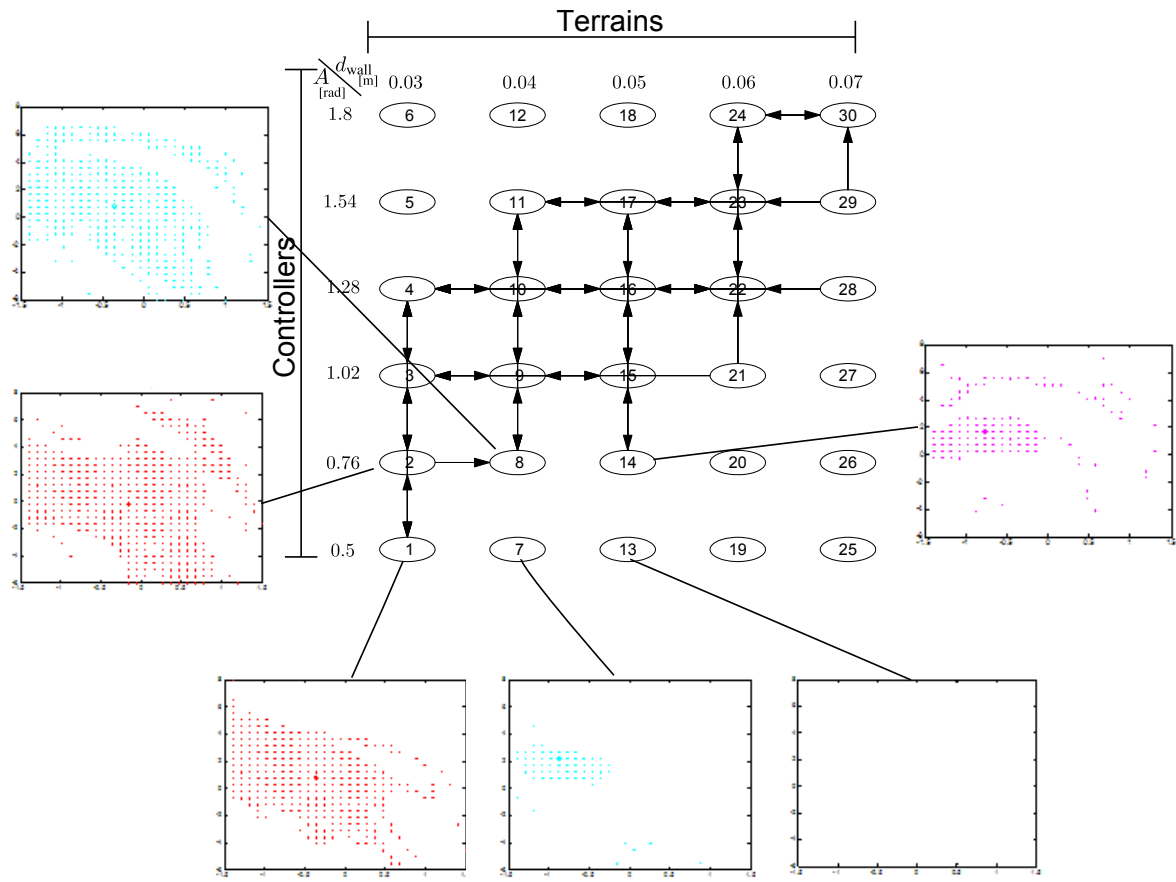
Figure 6.9: Transition graph of DSAC simulation. Six out of the 30 basins of attractions are shown in the figure.

143
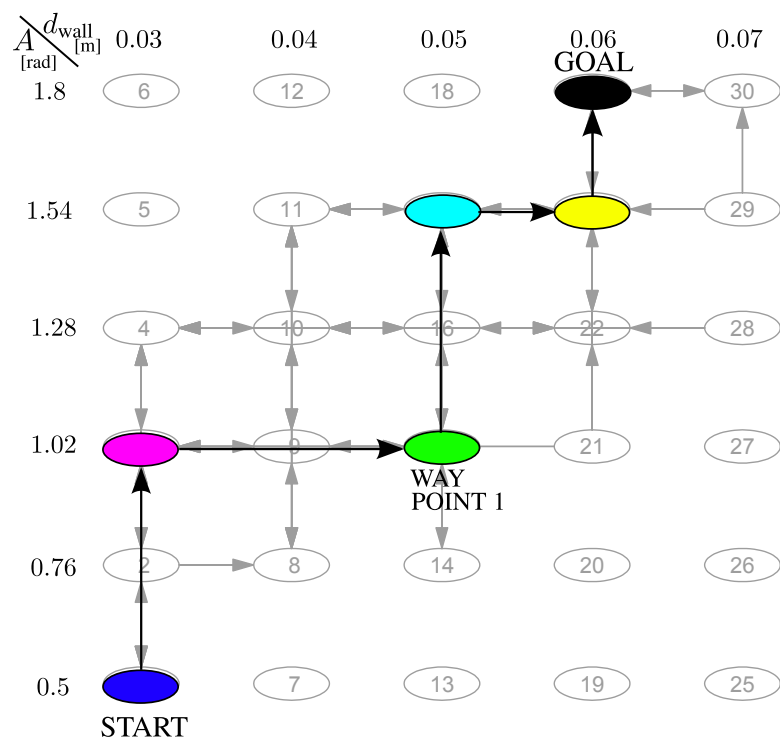
Figure 6.10: Planning DSAC simulation on the transition graph
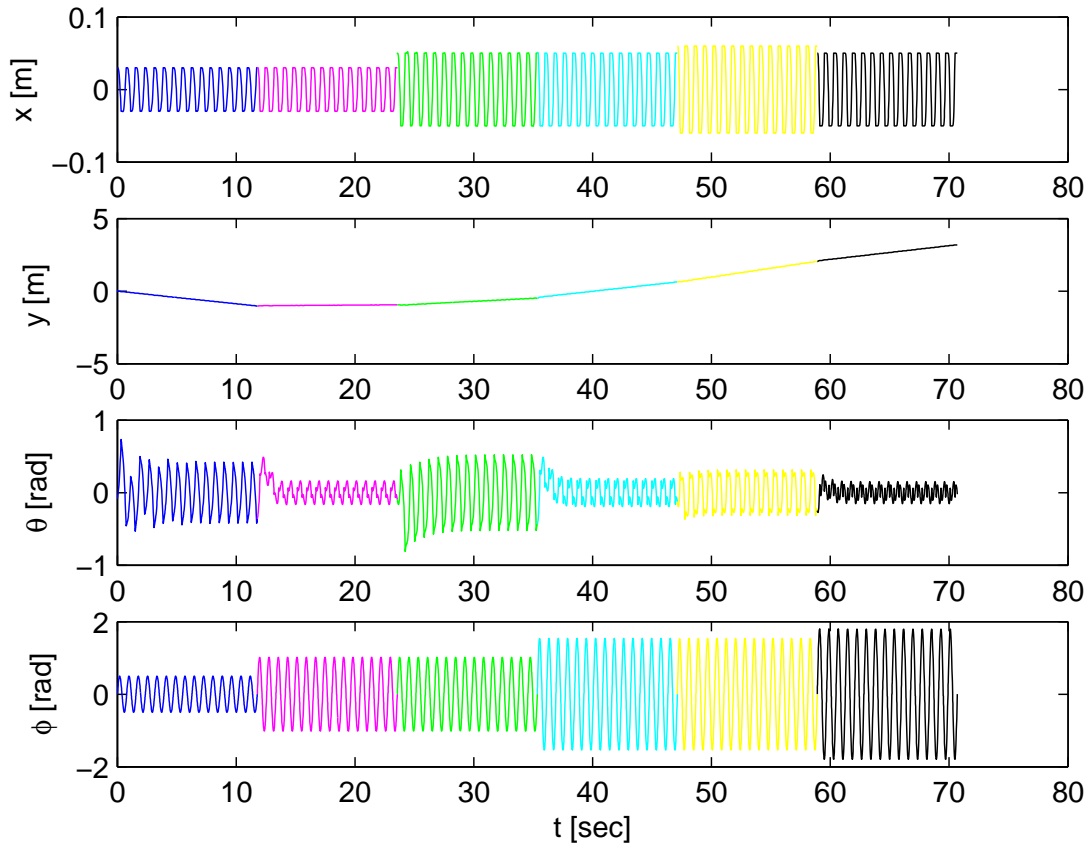
Figure 6.11: DSAC simulation on the transition graph

nism in fact moved downwards. Since we assume that the wall width can change independently of the height of the robot this did not form a problem. However, in a real climbing scenario, it is wise to use the height as a criteria for planning a path. This will be discussed in the future work chapter.

6.4 ParkourBot implementation

Similar to the DSAC implementation we have implemented the algorithm for the ParkourBot. This section will show the graph construction, planning and simulation. Furthermore, we will conclude with a proof of concept experiment showing a switch between three environments.

Four different wall width were chosen for this simulation ($d = 0.35$m, 0.5167m, 0.6833m, and 0.85m). Four different leg angles were chosen as possible controllers ($\psi_0 = 0.2$rad, 0.35rad, 0.5rad, and 0.65rad). The leg retraction, corresponding to energy input, is chosen to be $\zeta = 0.85$. All other mechanism parameters are the same as in Table 5.1 in Chapter 5. Using the cell mapping technique, the 30 basins of attractions were approximated. Using Algorithm 1, the adjacency matrix is constructed and converted into the transition graph, as shown in Figure 6.12.

A few notes about these basins of attraction and the transition graph:

1. The basins of attraction for $d = 0.35$m and $\psi_0 = 0.2$rad, 0.35rad are empty due to the fact that robot does not fit inside this wall width with this shallow leg angle. In order to climb such a small wall width the leg angles must be greater than 0.4rad.

2. Due to the ballistic motion of the ParkourBot in flight phase, transitioning to larger wall widths can only be done in small steps, e.g., with leg angle $\psi_0 = 0.35$rad, from wall width $d = 0.5167$m to $d = 0.6833$m and from $d = 0.6833$m to $d = 0.85$m. However, transitioning into narrower wall width can be done in larger step sizes, e.g., from $d = 0.85$m to $d = 0.5167$m. While increasing the wall width the velocity in the x remains constant however the velocity in the y direction increases in the negative direction. When wall width increases this change is greater than the change when the wall width decreases.

3. In simulation we assume that the leg never slips against the wall. In practice the leg does slip in high angled incoming velocity vector and large leg angles. Therefore, in practice leg angle will have an upper bound of approximately $\psi_0 < 0.5$rad.
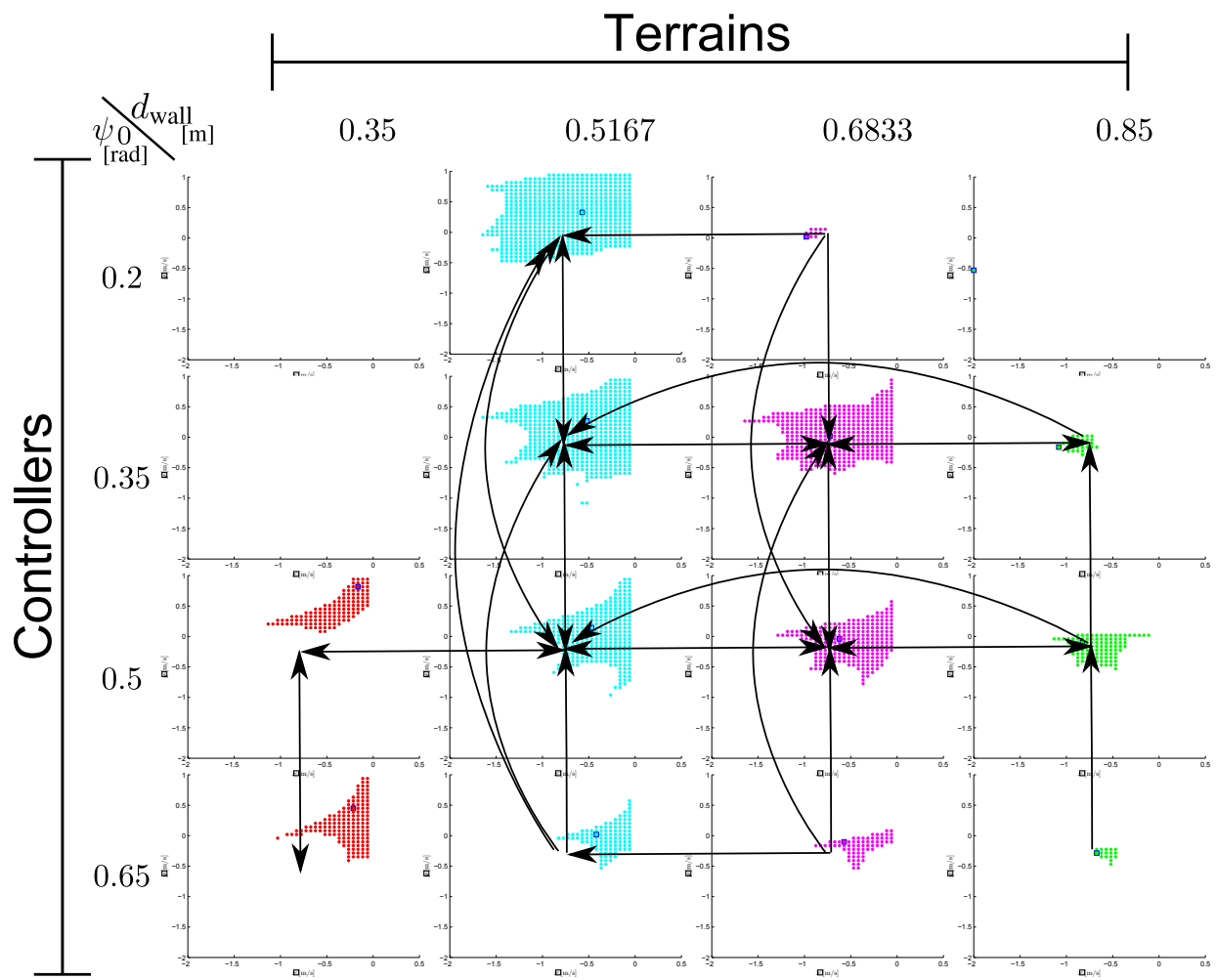
Figure 6.12: Transition graph of ParkourBot simulation

$\begin{array}{c}d_{\text{wall}}\\ \psi_0\quad[\text{m}]\\ [\text{rad}]\end{array}$	0.35	0.5167	0.6833	0.85
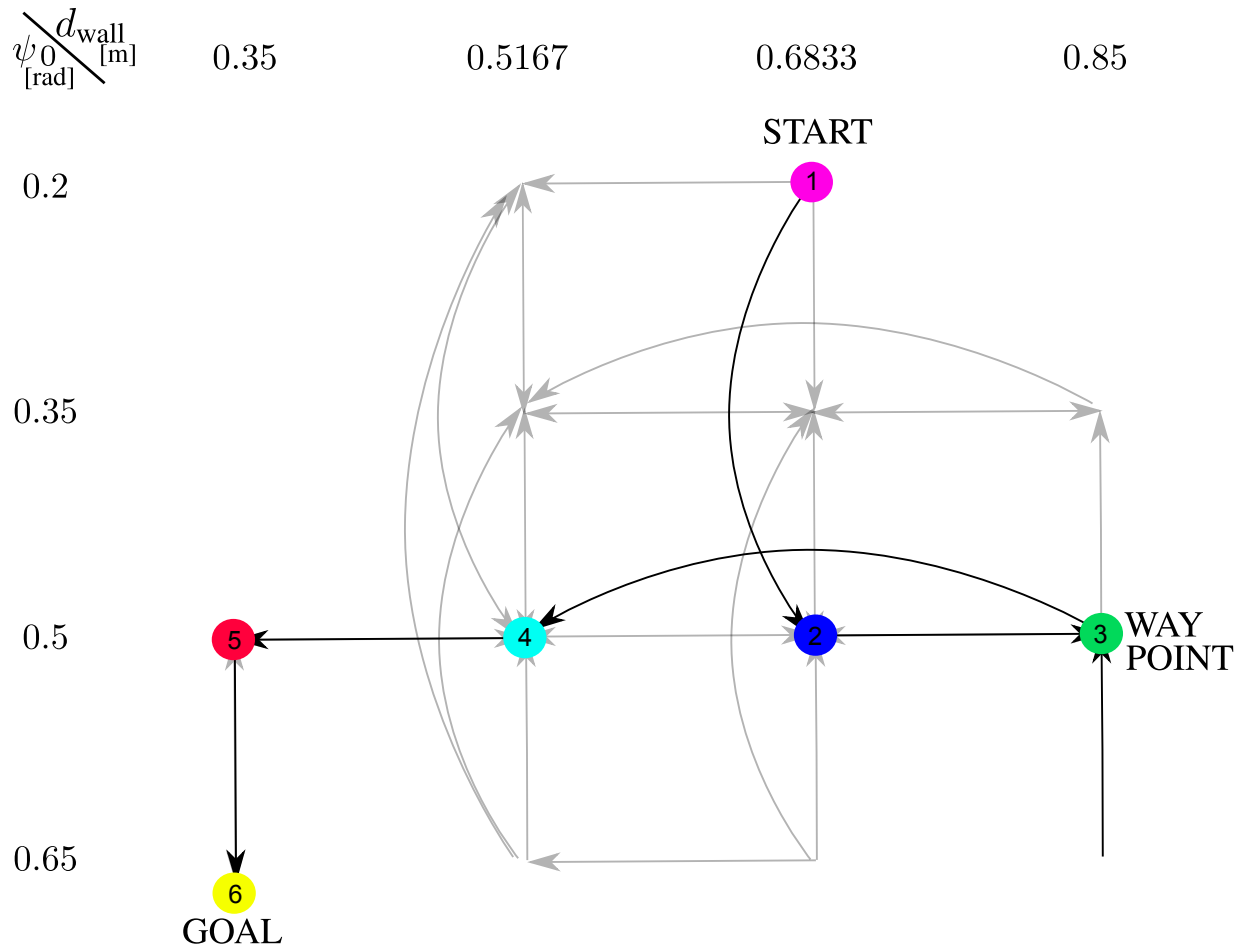
Figure 6.13: Planning ParkourBot simulation on the transition graph

As an example for the planning process we plan a path from the vertex with control $\psi_0 = 0.2$rad in $d = 0.6833$m to the goal at $\psi_0 = 0.65$rad in $d = 0.35$m via $\psi_0 = 0.5$rad in $d = 0.85$m (see Figure 6.13). Once again, this plan was obtained manually, however in a more complex graph a shortest path algorithm should be used. The simulation of this path is given in Figure 6.14. As can be seen, approximately three jumps are all that are needed after each change in control or environment to stabilize into the new attractor.
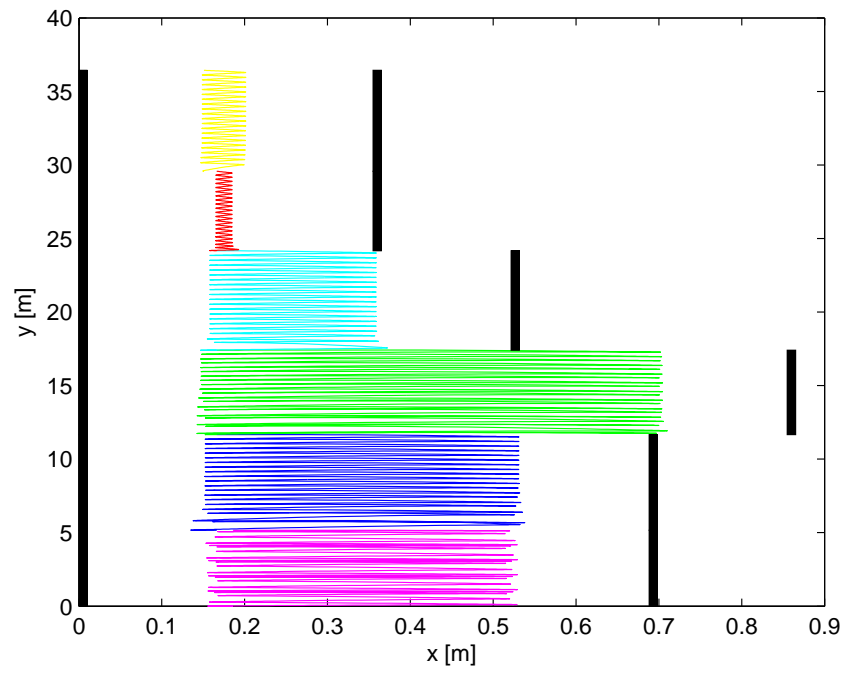
Figure 6.14: ParkourBot simulation on the transition graph

6.5 ParkourBot closed-loop experiment

We have conducted an experiment as a proof of concept of the technique described in this chapter. An identical setup to the one used in Chapter 5 was used. To simulate an environment change, we manually placed a wooden beam next to the wall to change the width. Three wall widths were chosen, $d = 0.456\text{m}, 0.502\text{m}, 0.54\text{m}$. Unlike the simulations previously shown, we used a gravitational acceleration of $g = 1.961\frac{\text{m}}{\text{s}^2}$ and leg retraction, corresponding to energy input of $\zeta = 0.80$. Three controllers were chosen $\psi_0 \approx 0.1\text{rad}, 0.2\text{rad}, 0.3\text{rad}$. The transition graph was obtained in simulations (Figure 6.15). We empirically tested to verify the accuracy of these simulated basins of attractions. The planned experiment was to start at the medium wall width ($d = 0.502\text{m}$), then transition to wide wall width ($d = 0.54\text{m}$) and finally to the narrowest wall width ($d = 0.456\text{m}$).

Figure 6.16 and Figure 6.17 show an actual experiment traversing these terrain switches. Figure 6.16 depicts the data from the tracking system of the center of the mechanism vs. time. Figure 6.17 depicts fives different snapshots at interesting times. The first three at the controller switches and the last two frames at the new terrains. This experiment was conducted successfully several times.
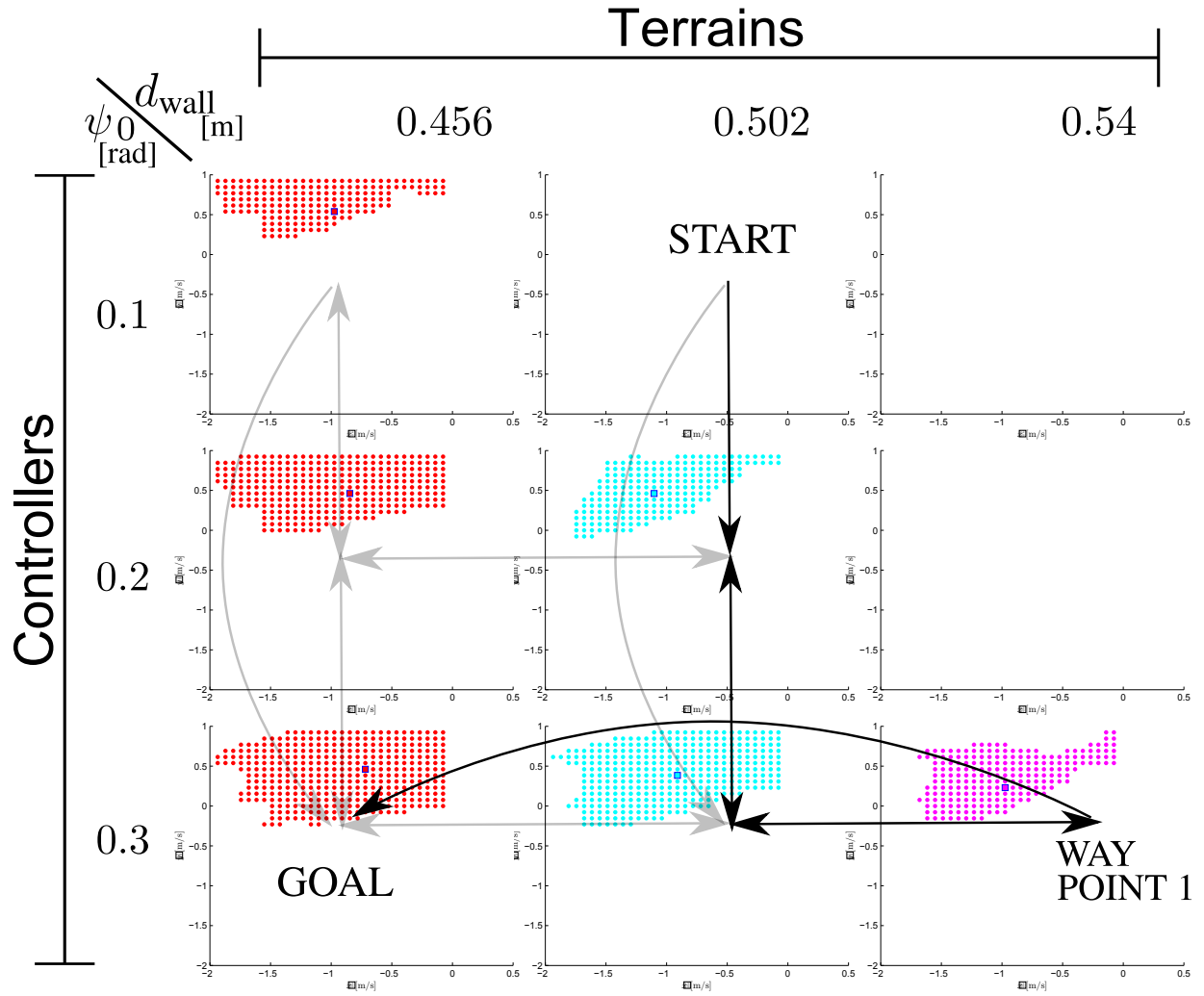
Figure 6.15: ParkourBot experiment - transition graph obtained by simulation and verified empirically.
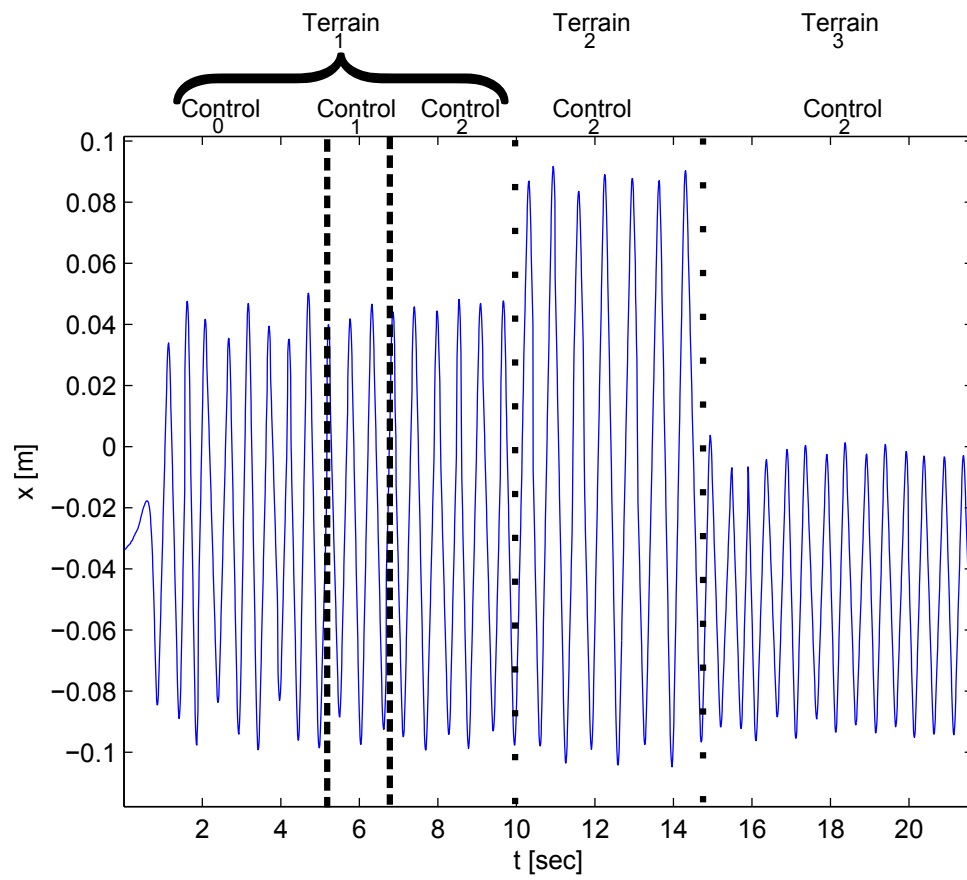
Figure 6.16: Data log of closed loop ParkourBot experiment.

152
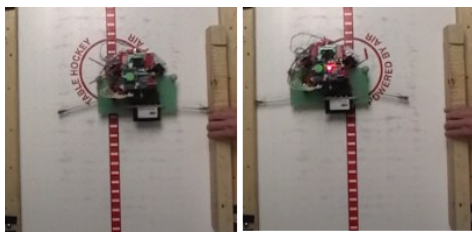
(a) 5.3s, Control 0: $\psi_0 = 0.1$rad, d=0.502m
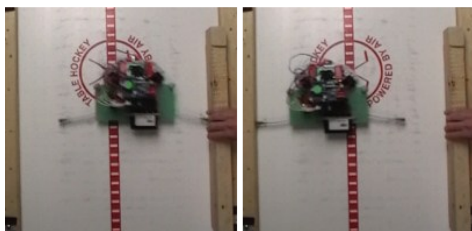
(b) 6s, Control 1: $\psi_0 = 0.2$rad, d=0.502m
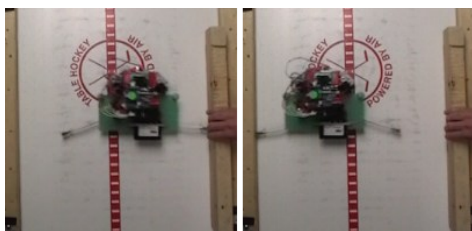
(c) 8s, Control 2: $\psi_0 = 0.3$rad, d=0.502m
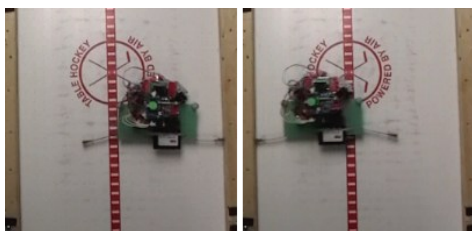
(d) 12s, Control 2: $\psi_0 = 0.3$rad, d=0.54m
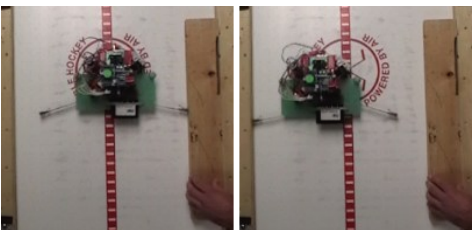
(e) 16s, Control 2: $\psi_0 = 0.3$rad, d=0.456m

Figure 6.17: ParkourBot experiment video snapshots.

153

Chapter 7

Future Work and Conclusions

7.1 Future Work

7.1.1 DSAC

The modeling in Chapter 3 assumes that only the distal end of the leg collides with the walls. This of course will not hold true for any arbitrary mechanism, control inputs and initial conditions. We intend to pursue the use of LCP formulation – linear complimentary problem (Stewart and Trinkle, 1996) to generalize the possible collision points with the wall and allow stick slip transitions. Moreover, we intend to relax our plastic impact assumption and use a more general impact model.

In Section 3.3 we observed that in some cases the non-symmetric climbing gait of the DSAC can be beneficial. The benefits were observed in the local stability, efficiency and climbing rate of the DSAC. In future work we intend to investigate other mechanisms such as bipedal robots to see if a similar phenomenon occurs. Moreover, investigating the hypothesis of the significance of limping in humans, specifically stroke victims, might lead to interesting findings.

7.1.2 DTAR

We have shown how the DTAR can climb robustly inside tubes. In problems such as search and rescue and biomedical applications, a common task requires pushing a scope through a tight opening, such as a scope in a collapsed building or a colonoscopy. Pushing these flexible scopes causes buckling, which might be reduced by vibratory excitation of the distal tip or of the entire mechanism. We intend to explore such applications for the DTAR. Moreover, building a wireless prototype, allowing the mechanism to twist while climbing and testing inside compliant tubes can further enhance the applicability of this mechanism.

7.1.3 From a tube climbing robot to a part orienting machine

The DTAR uses a simple eccentric mass rotation producing an oscillatory force which in turn generates the climbing motion inside tubes. What if we were to move the oscillation from the robot to the walls? What would happen if we remove the motor from the robot, making it a passive part, and oscillate the walls? Parts with high center of mass will move upwards while parts with low center of mass will move downwards. We conducted an initial proof of concept experiment shown in Figure 7.1 that explores this idea. A robot oscillates the walls on an inclined surface in a symmetric sinusoid. The experiment verified that parts with high center of mass move upwards while parts with low center of mass move downwards.

Unlike today's vibratory systems, where the locomotion originates from the asymmetry in the vibration, this method has symmetric environment and the asymmetry originates from the non-uniform mass distribution of the parts themselves. This method might therefore be useful for part orienting. Oscillatory walls can be placed inside a bin full of identical parts in arbitrary orientations. This setup should be able to pick and locomote parts with high CoM, which in turn forms a part orienting system. This scheme may be beneficial in smaller scales and even MEMS scaled parts.
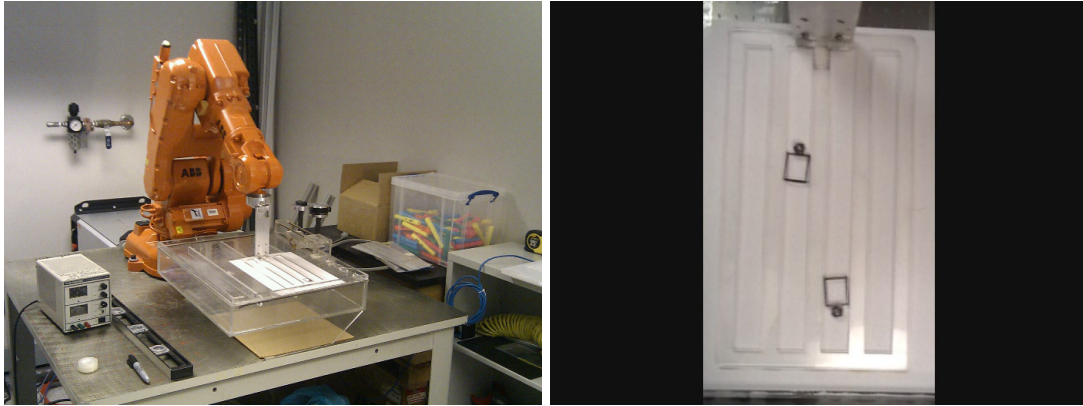
(a) Part orienting experiment setup   (b) Top view

Figure 7.1: Part orienting machine – proof of concept experiment. A robot oscillates the walls on an inclined surface in a symmetric sinusoid. Parts with high center of mass move upwards while parts with low center of mass move downwards.

7.1.4   ParkourBot

In order to overcome the problem of pitch change during flight phase, we have implemented a gyro-stabilizer which increases the inertia of the system to reduce angle rotation. However, the use of the gyro-stabilizer might not be enough to overcome rotations during impacts resulting in high torques on the body. This will mostly occur when leg angles are very high or very low, since the mechanism was designed to make the leg force pass exactly through the CoM only at $30°$ leg angles. We have also added an active gimbal that can be torqued to enable active pitch control. This too is limited to the point where the gyro hits the gimbal limits. To overcome these problems we intend to allow active movement of the CoM. If done correctly in conjunction with leg angle change, the torque during impact can be reduced considerably.

7.1.5 Closed-loop

In Chapter 3, we have shown how a change in leg length ratio (γ) can alter periodicity and can result in a period doubling cascade in an open-loop system. In the experimental setup, the leg angle was changed at the beginning of each experiment. Theoretically, since the leg mass is negligible, the leg length can be changed during flight phase without altering the dynamics of the system. This parameter change can be used as another control input in our closed-loop algorithm.

In the closed-loop algorithm in Chapter 6 we have recorded the attractor and its basin of attraction, for a number of different controls and environments. The algorithm then found the valid transitions of controls to arrive at a controller enabling stable transitions to different terrains. In our setup, the change in wall width occurs independently of the vertical location of the mechanisms. That is, even if the climber climbed downwards but has stabilized into its attractor, the next wall width change will be executed. We will like to relax this assumption and allow more natural change in wall width such as a static vertical walls with different wall widths. We intend to take into account the climbing rate of each attractor and use this information in the planning stage. Moreover we will use the estimation of the number of steps needed to converge to the attractor.

In many systems, and specifically in the ParkourBot system, we have observed in simulation a meta stable motion. In our case these were the period-4 climbing gaits. The planning technique described in this this thesis did not allow transition to these attractors since they were not stable. However, in reality the robot was able to leap approximately 20 times before an imminent crash. This hints that even these non-stable solutions might be beneficial for exploring more planning possibilities.

The authors in (Seyfarth et al., 2003) have shown how a simple horizontal SLIP bipedal running model can have a "dead-beat" controller with a simple leg swing. In their work, the Poincaré map from apex to next apex can be stabilized to a desired height in one step by starting

to rotate the leg once it reaches apex. We have started investigating the option of using a similar technique in our ParkourBot. Figure 7.2 shows a preliminary simulation showing that indeed a similar simple leg sweep can stabilize the mechanism while changing wall width from 54cm to 48cm to 60cm. This method should first be proven and verified experimentally. Moreover, this method is bounded in its wall width variation per initial leg angle and sweeping velocity. We can still employ a similar funneling technique as shown in Chapter 6 to enable larger wall width variations.
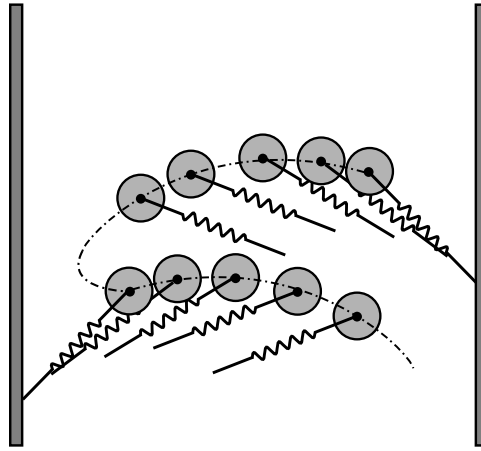
(a) Leg swing schematics
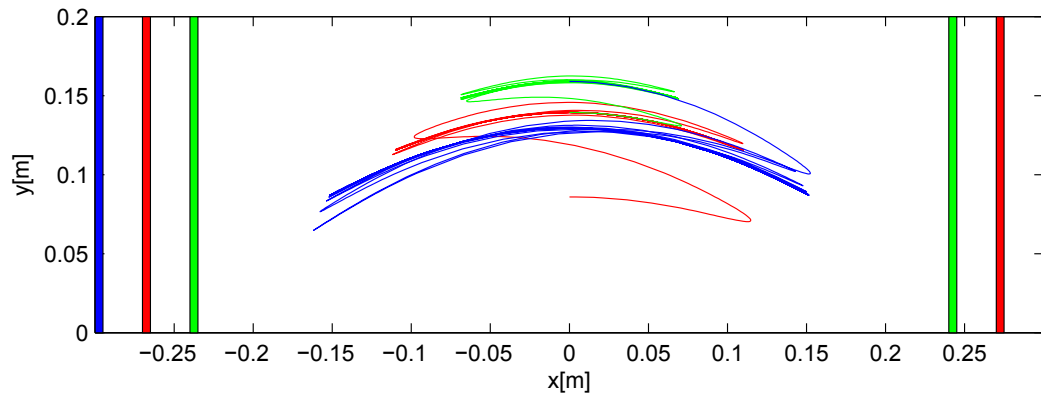
(b) Simulation results varying wall width

Figure 7.2: Using leg swing method. Jumping in place varying wall width from 54cm to 48cm to 60cm.

7.2 Conclusions

The family of mechanisms explored in this thesis aims to perform stable climbing with minimal design and control complexities. Unique to these mechanisms is the use of dynamic motions to achieve this goal. As it stands, these mechanisms can achieve a simple climbing task with simple design and without any complex controls.

The dynamic motions uniquely used by these mechanisms entail the potential not only to climb with a low number of actuators, but also to accomplish more complex tasks, such as overcoming obstacles, and climbing quickly and efficiently. Even in open-loop, the mechanisms described here, can robustly climb relatively wide range of wall gaps.

To deal with more versatile terrain, we incrementally added complexity to these minimalistic mechanisms. In particular, we have shown how *a priori* knowledge of the terrain allows the DSAC and the ParkourBot to switch between controllers to enable a safe transition of terrains. This method uses the knowledge gained in the open-loop investigations to find a valid transition of controls.

One might ask why minimalism is even beneficial. Motors are becoming cheaper, therefore, why should we want simple design with only a single motor and no complex control? Although true in some respects, minimalism still provides a huge advantage in others. One example is when a mechanism is shrunk to small scales. In such cases, complex elements including springs, bearings, and linear motors are almost impossible to package. However, a mechanism which can achieve the task using a single revolute motor can be relatively easily miniaturized. This was initially used in the development of the DTAR mechanism, which we intend to miniaturize even further.

Another way to appreciate the advantages of minimalism is to consider the difference in speed between running robots and biological runners. Biological runners are several times faster than the fastest robotic runner. One reason for this difference is the use of the self stability

properties in biological locomotion. By using the self stability of the natural system, the need for fast online feedback is reduced. This motivates the use of self stable (open-loop) mechanisms in a minimalistic approach as discussed here.

To summarize, the family of dynamic mechanisms analyzed in this thesis serves to exemplify how dynamic motions can help design mechanisms in a minimalistic way which can achieve results comparable to other mechanisms with more complex design and control. We have analyzed the open-loop stability characteristic of the two link DSAC, the miniature tube climbing DTAR and the spring legged ParkourBot. We have shown how varying control parameters in these systems changes the climbing motion crucial for their stability. Finally, using these open-loop stability characteristics, and adding minimal complexity in control enabled the mechanisms to traverse more complex environments.

Appendix A

Nondimensionalizing

A.1   Nondimensionalizing differential equations

The use of nondimensionalizing equations of motion is a practical tool in analyzing dynamical systems. This method can reduce the number of parameters and introduce important nondimensional (unitless) ratios instead of specific parameters.

Nondimensionalisation scales each variable, dependent and independent, by a characteristic value which results in a nondimensional variable.

This methods has several uses:

1. It creates dimensionless parameters which are ratios of the the differential equation parameters.

2. Since the coefficients of the differential equation are dimensionless it allows to compare terms and find the dominant versus negligible terms.

3. Gives intuition of what should be varied in an experimental setup.

4. Can reduce the number of parameters by up to the number of fundamental units involved in the equation. In our case of mechanical system, this procedure can reduce by up to three

parameters corresponding to the following fundamental units: mass, length, and time.

The procedure in nondimensionalizing a differential equation is

1. List all variables including dependant and independent variables.

2. For each variable choose a characteristic parameter in order to define a new, nondimensional variable. As a simple example we will nondimensionalize the differential equation of a damped mass oscillator with an external force F (Figure A.1).

$$m\frac{d^2x}{dt^2} + c\frac{dx}{dt} + kx = F,$$

where x is the displacement of the mass, c is the damping coefficient, and k is the spring constant. For the variable x (with units of length $[m]$) one might choose a general characteristic length (l_0). The new nondimensional variable will be $\bar{x} = \frac{x}{l_0}$. In case we are nondimensionalizing a differential equation, we must also nondimensionalize time, in our example t_0. There is no unique way to choose the characteristic parameters. However, different choices will change the ratios.

3. Rewrite the differential equations with the new nondimensional parameters. Note that in order to rewrite the derivatives w.r.t time one must use the chain rule. For example: for an x variable ($[m]$) and time ($[sec]$), let us choose characteristic length and time such that the new nondimensional variable are: $\bar{x} = \frac{x}{l_0}$ and $\bar{t} = \frac{t}{t_0}$. Now the first time derivative of x w.r.t time is:

$$\frac{dx}{dt} = \frac{d\bar{x}l_0}{d\bar{t}}\frac{d\bar{t}}{dt} = l_0\frac{d\bar{x}}{d\bar{t}}\frac{1}{t_0} = \frac{l_0}{t_0}\frac{d\bar{x}}{d\bar{t}}.$$

Similarly for the accelerations

$$\frac{d^2x}{dt^2} = \frac{l_0}{t_0{}^2}\frac{d^2\bar{x}}{d\bar{t}^2}.$$

The derivative is now fully dimensionless and the units are all "outside" of the derivative. In our example these units are all units of force. The nondimensional differential equation

164
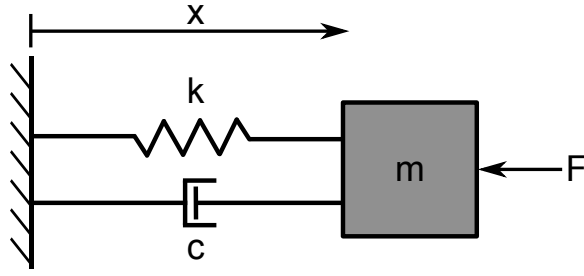
Figure A.1: Schematics of damped mass oscillator.

is now converted to

$$m\frac{l_0}{t_0{}^2}\frac{d^2\bar{x}}{d\bar{t}^2} + c\frac{l_0}{t_0}\frac{d\bar{x}}{d\bar{t}} + kl_0\bar{x} = F,$$

4. Since all the units are now in the coefficient of the differential equation, dividing the equations by one of these coefficients will normalize the equations and yield nondimensional coefficients (or parameters). In our example we will divide by the leftmost coefficient

$$\frac{d^2\bar{x}}{d\bar{t}^2} + \frac{ct_0}{m}\frac{d\bar{x}}{d\bar{t}} + \frac{kt_0{}^2}{m}\bar{x} = \frac{Ft_0{}^2}{ml_0},$$

We are now free to choose what the characteristic length and time (l_0 and t_0) should be. By picking $l_0 = \frac{F}{k}$, and $t_0 = \sqrt{\frac{m}{k}}$, we arrive at the final nondimensional equation

$$\frac{d^2\bar{x}}{d\bar{t}^2} + \frac{c}{\sqrt{km}}\frac{d\bar{x}}{d\bar{t}} + \bar{x} = 1.$$

This reduced the number of parameters from four (m, c, k, and F) to one ($\frac{c}{\sqrt{km}}$). As was explained above, this is a reduction by the number of fundamental units, in our case three - time, mass, and length.

Appendix B

Equations of Motion

B.1 General equations of motion

This section will derive the general equations of motion of the three phases.

B.1.0.1 Free flight phase

Using the Lagrange method the energy must be first found. For that, the kinematics including the position of the two masses, then velocities are found - see Figure B.1 for symbols.

$$r_{m_1} = \begin{bmatrix} x + b_1 \sin\theta \\ y - b_1 \cos\theta \end{bmatrix}, \quad r_{m_2} = \begin{bmatrix} x + b_1 \sin\theta - b_2 \sin(\theta + \phi) \\ y - b_1 \sin\theta + b_2 \sin(\theta + \phi) \end{bmatrix} \tag{B.1}$$

where x, y, θ and ϕ are time dependant, i.e., $x(t)$, $y(t)$, $\theta(t)$ and $\phi(t)$.

Velocities of masses:

$$v_{m_1} = \frac{dr_1}{dt} = \begin{bmatrix} \dot{x} + b_1 \cos\theta\,\dot{\theta} \\ \dot{y} + b_1 \sin\theta\,\dot{\theta} \end{bmatrix}, \quad v_{m_2} = \begin{bmatrix} \dot{x} + b_1 \cos\theta\,\dot{\theta} - b_2 \cos(\theta - \phi)(\dot{\theta} - \dot{\phi}) \\ \dot{y} + b_1 \sin\theta\,\dot{\theta} - b_2 \sin(\theta - \phi)(\dot{\theta} - \dot{\phi}) \end{bmatrix} \tag{B.2}$$
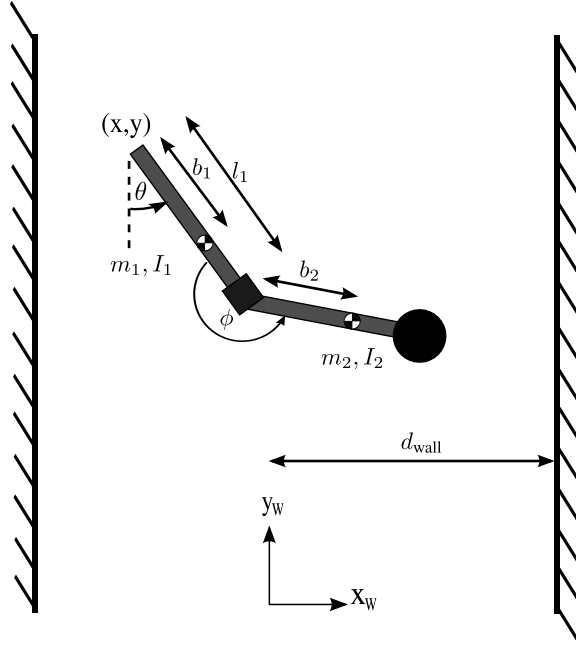
Figure B.1: Schematics of two link mechanism climbing between two parallel walls.

The Lagrangian is written as $L = T - V$ where the kinetic and potential energies are[1]:

$$T = \frac{1}{2}\left(m_1 v_{m_1}^2 + m_2 v_{m_2}^2 + I_1 \dot{\theta}^2 + I_2(\dot{\theta} + \dot{\phi})^2\right) =$$
$$\frac{1}{2}\left(I_1 \dot{\theta}^2 + m_1\left(\left(\dot{x} + b_1 \cos\theta\dot{\theta}\right)^2 + \left(\dot{y} + b_1 \sin\theta\dot{\theta}\right)^2\right) + I_2\left(\dot{\theta} + \dot{\phi}\right)^2 +$$
$$m_2\left(\left(\dot{x} + l_1 \cos\theta\dot{\theta} - b_2 \cos\theta + \phi\left(\dot{\theta} + \dot{\phi}\right)\right)^2 + \left(\dot{y} + l_1 \sin\theta\dot{\theta} - b_2 \sin\theta + \phi\left(\dot{\theta} + \dot{\phi}\right)\right)^2\right)\right)$$

(B.3)

$$V = -m_1 g r_{m_1} - m_2 g r_{m_2} = g\left(-\left(b_1 m_1 + l_1 m_2\right)\cos\theta + b_2 m_2 \cos\left(\theta + \phi\right) + \left(m_1 + m_2\right) y\right).$$

(B.4)

Next, Eq. 2.1 from Sec 2.1is used to find the equations of motion

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = 0$$

(B.5)

[1]We note that not all parameters are needed to describe this lagrangian. The number of independent geometric and mass parameters in the lagrangian can be reduced by two (c.f. Dullin (1994)). From the seven original parameters (m_1, m_2, l_1, I_1, I_2, b_1, and b_2) to five new parameters ($I_1 + m_2 l_1^2$, I_2, $m_2 b_2 l_1$, $m_1 b_1 + m_2 l_1$, and $m_2 b_2$).

In matrix form

$$M(q)\ddot{q} + h(q, \dot{q}) = 0, \tag{B.6}$$

where

$$M(q) = \begin{pmatrix} M_{11} & 0 & M_{13} \\ 0 & M_{22} & M_{23} \\ M_{31} & M_{32} & M_{33} \end{pmatrix}, \tag{B.7}$$

$$M_{11} = M_{22} = m_1 + m_2,$$

$$M_{13} = M_{31} + (b_1 m_1 + l_1 m_2) \cos\theta - b_2 m_2 \cos(\theta + \phi),$$

$$M_{23} = M_{32} = (b_1 m_1 + l_1 m_2) \sin\theta - b_2 m_2 \sin(\theta + \phi), \tag{B.8}$$

$$M_{33} = I_1 + I_2 + b_1^2 m_1 + (b_2^2 + l_1^2) m_2 - 2 b_2 l_1 m_2 \cos\phi.$$

and

$$h(q, \dot{q}) = \begin{pmatrix} -b_1 m_1 \sin\theta\, \dot{\theta}^2 + m_2(-l_1 \sin\theta\, \dot{\theta}^2 + b_2 \sin(\theta - \phi)(\dot{\theta} + \dot{\phi})^2 - b_2 \cos(\theta + \phi)\ddot{\phi} \\ g(m_1 + m_2) + b_1 m_1 \cos\theta\, \dot{\theta}^2 + m_2(l_1 \cos\theta\, \dot{\theta}^2 - b_2 \cos(\theta + \phi)(\dot{\theta} + \dot{\phi})^2 + b_2 \sin(\theta + \phi)\ddot{\phi}) \\ g(b_1 m_1 + l_1 m_2) \sin\theta - g b_2 m_2 \sin(\theta + \phi) + I_2 \ddot{\phi} + b_2 m_2(l_1 \sin\phi(2\dot{\theta} + \dot{\phi})\dot{\phi} + (b_2 - l_1 \cos\phi)\ddot{\phi}) \end{pmatrix}$$

$$\tag{B.9}$$

B.1.0.2 Impact phase

From Sec: 3.1.2.2, conservation of angular momentum around the contact point during impact is used to calculate the state after impact

$$
\begin{aligned}
\dot{\theta}^+ = {} & \frac{1}{I_1 + I_2 + b_1{}^2 m_1 + l_1 + b_2{}^2 m_2 - 2 l_1 b_2 m_2 \cos\theta^-} \\
& \cdot \Bigg(\left((b_1 m_1 + l_1 m_2) \cos\theta - b_2 m_2 \cos(\theta^- + \phi^-)\right) \dot{x}^- \\
& \quad + \left((b_1 m_1 + l_1 m_2) \sin\theta^- - b_2 m_2 \sin(\theta^- + \phi^-)\right) \dot{y}^- \\
& \quad + (I_1 + I_2 + b_1{}^2 m_1 + b_2{}^2 m_2 + l_1{}^2 m_2 - 2 b_2 l_1 m_2 \cos\phi^-)\dot{\theta}^- \\
& \quad + (I_2 + b_2{}^2 m_2 - l_1 b_2 m_2 \cos\phi^-)(\dot{\phi}^- - \dot{\phi}^+) \Bigg)
\end{aligned}
\tag{B.10}
$$

169

This last equation can further be simplified by noting that since ϕ is constraint, $\dot{\phi}^- - \dot{\phi}^+ = 0$, and $\dot{\theta}^+$ reduces to

$$\dot{\theta}^+ = \frac{1}{I_1 + I_2 + b_1{}^2 m_1 + l_1 + b_2{}^2 m_2 - 2l_1 b_2 m_2 \cos\theta^-}$$
$$\cdot \Big(\big((b_1 m_1 + l_1 m_2) \cos\theta - b_2 m_2 \cos(\theta^- + \phi^-) \big) \dot{x}^-$$
$$+ \big((b_1 m_1 + l_1 m_2) \sin\theta^- - b_2 m_2 \sin(\theta^- + \phi^-) \big) \dot{y}^-$$
$$+ (I_1 + I_2 + b_1{}^2 m_1 + b_2{}^2 m_2 + l_1{}^2 m_2 - 2b_2 l_1 m_2 \cos\phi^-) \dot{\theta}^- \Big). \tag{B.11}$$

using this and the flip of coordinates during impact, the entire impact phase map is constructed

$$\begin{bmatrix} x^+ \\ y^+ \\ \theta^+ \\ \dot{x}^+ \\ \dot{y}^+ \\ \dot{\theta}^+ \end{bmatrix} = \begin{bmatrix} -x^- \\ y^- \\ -\theta^- \\ 0 \\ 0 \\ -\dot{\theta}^+ \end{bmatrix}, \tag{B.12}$$

where $\dot{\theta}^+$ is calculated using Eq. B.11.

B.1.0.3 Stance phase

Using the process described in Section 3.1.2.3, one can find the equations of motion of the mechanism and the contact forces from the wall. The problem can be decoupled when the leg is in contact with the wall, while keeping the no rebound, no slip assumption. Only the equations of motion for the $\theta, \dot{\theta}$ must be solved while observing the contact forces to see when they change sign, corresponding to transition to flight phase. The equation of motion for $\theta, \dot{\theta}$ is the last (third) row of Eq. B.6

$$\left(I_1 + I_2 + b_1^2 m_1 + (b_2^2 + l_1^2)m_2 - 2b_2 l_1 m_2 \cos\phi\right)\ddot\theta + g(b_1 m_1 + l_1 m_2)\sin\theta$$

$$- gb_2 m_2 \sin(\theta + \phi) + I_2\ddot\phi + b_2 m_2(l_1 \sin\phi(2\dot\theta + \dot\phi)\dot\phi + (b_2 - l_1 \cos\phi)\ddot\phi = 0, \quad \text{(B.13)}$$

The contact force is calculated using the Lagrange multipliers method. As in Eq. 3.9.

$$\lambda_{ext}(q, \dot{q}) = \left(A(q)M(q)^{-1}A(q)^T\right)^{-1}\left(\dot{A}(q)\dot{q} - A(q)M(q)^{-1}h(q, \dot{q})\right), \quad \text{(B.14)}$$

where $A(q) = \left(\begin{smallmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \end{smallmatrix}\right)$ and $M(q)$ and $h(q, \dot{q})$ are from Eq. B.7 and B.9.

B.2 Nondimensionalisation of the equations of motion

The conversion to nondimensional equations of motion is done by first picking the characteristic length and time. The characteristic length and time for this non-unique set are d_{wall} and $\frac{1}{\omega}$, respectively. The non-dimensional variables are then converted to

$$x^* = \frac{x}{d_{\text{wall}}}, \quad y^* = \frac{y}{d_{\text{wall}}}, \quad \theta^* = \theta, \quad \phi^* = \phi, \quad \tau = \omega t. \quad \text{(B.15)}$$

where $[.]^*$ represents the nondimensional variable and τ is the nondimensional time.

all the dimensional variables are replaced with their non-dimensional counterparts. Switching the configuration variables is trivial, e.g., $x = x^* d_{\text{wall}}$. In order to find the conversion for the velocities and acceleration the chain rule is used.

$$\begin{aligned}
\frac{dx}{dt} &= \frac{dx}{d\tau}\frac{d\tau}{dt} = d_{\text{wall}}\frac{dx^*}{d\tau}\frac{d\tau}{dt} = \omega\, d_{\text{wall}}\frac{dx^*}{d\tau} \\
\frac{dy}{dt} &= \frac{dy}{d\tau}\frac{d\tau}{dt} = d_{\text{wall}}\frac{dy^*}{d\tau}\frac{d\tau}{dt} = \omega\, d_{\text{wall}}\frac{dy^*}{d\tau} \\
\frac{d\theta}{dt} &= \frac{d\theta}{d\tau}\frac{d\tau}{dt} = \frac{d\theta^*}{d\tau}\frac{d\tau}{dt} = \omega\frac{d\theta^*}{d\tau} \\
\frac{d\phi}{dt} &= \frac{d\phi}{d\tau}\frac{d\tau}{dt} = \frac{d\phi^*}{d\tau}\frac{d\tau}{dt} = \omega\frac{d\phi^*}{d\tau}
\end{aligned} \quad \text{(B.16)}$$

Similarly the acceleration variables can be found

$$\begin{aligned}
\frac{d^2x}{dt^2} &= \omega^2\, d_{\text{wall}}\frac{d^2x^*}{d\tau^2} \\
\frac{d^2y}{dt^2} &= \omega^2\, d_{\text{wall}}\frac{d^2y^*}{d\tau^2} \\
\frac{d^2\theta}{dt^2} &= \omega^2\frac{d^2\theta^*}{d\tau^2} \\
\frac{d^2\phi}{dt^2} &= \omega^2\frac{d^2\phi^*}{d\tau^2}
\end{aligned}$$

(B.17)

After replacing the dimensional variables of Eq. B.6 with the nondimensional variables from Eqs. B.15, B.16, and B.17 the equations should be divided by one of the coefficients in order to normalize the equations. For simplicity we will normalize by $M(1,1)$. Since the third equation (corresponding to θ having units of $\frac{1}{s^2}$) is also divided by $M(1,1)$ (with units of $\frac{m}{s^2}$), we need to also divide this equation by the characteristic length. After these replacement and normalization we have the nondimensional matrix form of the general flight phase equations of motion

$$M^*(q^*)\ddot{q}^* + h^*(q^*,\dot{q}^*) = 0, \tag{B.18}$$

where

$$M^*(q^*) = \begin{pmatrix} 1 & 0 & M^*_{13} \\ 0 & 1 & M^*_{23} \\ M^*_{31} & M^*_{32} & M_{33} \end{pmatrix}, \tag{B.19}$$

and

$$\begin{aligned}
M^*_{13} &= M^*_{31} + \left(\frac{\delta\beta}{1+\mu} + \frac{\delta\mu}{1+\mu}\right)\cos\theta - \frac{\beta\gamma\mu\delta}{(1+\mu)}\cos(\theta+\phi), \\
M^*_{23} &= M^*_{32} = \left(\frac{\delta\beta}{1+\mu} + \frac{\delta\mu}{1+\mu}\right)\sin\theta - \frac{\beta\gamma\mu\delta}{(1+\mu)}\sin(\theta+\phi), \\
M^*_{33} &= \rho_1 + \rho_2 + \frac{\delta^2\beta^2}{1+\mu} + \left(\frac{\gamma^2\mu\delta^2\beta^2}{1+\mu} + \frac{\delta^2\mu}{1+\mu}\right) - 2\frac{\gamma\mu\delta^2\beta}{1+\mu}\cos\phi,
\end{aligned} \tag{B.20}$$

and

172

$$h^*(q^*, \dot{q}^*) = \begin{pmatrix} -\dfrac{\delta\beta}{1+\mu}\sin\theta\,\dot\theta^2 + m_2(-\dfrac{\delta\mu}{1+\mu}\sin\theta\,\dot\theta^2 + \dfrac{\beta\gamma\mu\delta}{(1+\mu)}\sin(\theta-\phi)(\dot\theta+\dot\phi)^2 - \dfrac{\beta\gamma\mu\delta}{(1+\mu)}\cos(\theta+\phi)\ddot\phi) \\[2mm] \Omega + \dfrac{\delta\beta}{1+\mu}\cos\theta\,\dot\theta^2 + (\dfrac{\delta\mu}{1+\mu}\cos\theta\,\dot\theta^2 - \dfrac{\beta\gamma\mu\delta}{(1+\mu)}\cos(\theta+\phi)(\dot\theta+\dot\phi)^2 + \dfrac{\beta\gamma\mu\delta}{(1+\mu)}\sin(\theta+\phi)\ddot\phi) \\[2mm] \Omega(\dfrac{\delta\beta}{1+\mu}+\dfrac{\delta\mu}{1+\mu})\sin\theta - \Omega\dfrac{\beta\gamma\mu\delta}{(1+\mu)}\sin(\theta+\phi) + \rho_2\ddot\phi + ... \\[2mm] ...(\dfrac{\gamma\mu\delta^2\beta}{1+\mu}\sin\phi(2\dot\theta+\dot\phi)\dot\phi + (\dfrac{\gamma^2\mu\delta^2\beta^2}{1+\mu} - \dfrac{\gamma\mu\delta^2\beta}{1+\mu}\cos\phi)\ddot\phi \end{pmatrix},$$

$$\text{(B.21)}$$

where the nondimensional parameters are

$$\mu = \frac{m_2}{m_1}, \beta = \frac{b_1}{l_1}, \gamma = \frac{b_2}{b_1}, \delta = \frac{l_1}{d_{\text{wall}}}, \Omega = \frac{g}{\omega^2 d_{\text{wall}}}, A,$$

$$\rho_1 = \frac{I_1}{d_{\text{wall}}^2(m_1+m_2)}, \quad \rho_2 = \frac{I_2}{d_{\text{wall}}^2(m_1+m_2)}.$$

$$\text{(B.22)}$$

Notice that these nondimensional parameters are not unique, we could have chosen other parameters such as $\frac{m_1}{m_2}$.

Using this method the nondimensional equations for the impact phase are

$$\dot\theta^{*+} = \frac{1}{\left((\rho_1+\rho_2) + \frac{\delta^2\beta^2}{1+\mu} + \frac{\gamma^2\mu\delta^2\beta^2}{1+\mu} + \frac{\mu\delta^2}{1+\mu} - 2\frac{\gamma\mu\delta^2\beta}{1+\mu}\cos\phi^*\right)}$$

$$\cdot\left(\frac{\delta\beta}{1+\mu}\cos\theta^*\dot x^* + \frac{\delta\mu}{1+\mu}\cos\theta^*\dot x^* - \frac{\gamma\mu\delta\beta}{1+\mu}\cos\theta^* + \phi^*\dot x^* +\right.$$

$$\frac{\delta\beta}{1+\mu}\sin\theta^*\dot y^* + \frac{\delta\mu}{1+\mu}\sin\theta^*\dot y^* -$$

$$\frac{\gamma\mu\delta\beta}{1+\mu}\sin\theta^* + \phi^*\dot y^* + \rho_1\dot\theta^{*-} + \rho_2\dot\theta^{*-} + \frac{\delta^2\beta^2}{1+\mu}\dot\theta^{*-} +$$

$$\left.\frac{\gamma^2\mu\delta^2\beta^2}{1+\mu}\dot\theta^{*-} + \frac{\mu\delta^2}{1+\mu}\dot\theta^{*-} - 2\frac{\gamma\mu\delta^2\beta}{1+\mu}\cos\phi^*\theta^*\right) \quad \text{(B.23)}$$

Finally, the nondimensional equation of motion for the stance phase is

$$\left(\rho_1 + \rho_2 + \frac{\delta^2\beta^2}{1+\mu} + (\frac{\gamma^2\mu\delta^2\beta^2}{1+\mu} + \frac{\delta^2\mu}{1+\mu}) - 2\frac{\gamma\mu\delta^2\beta}{1+\mu}\cos\phi\right)\ddot\theta^* +$$

$$\Omega(\frac{\delta\beta}{1+\mu} + \frac{\delta\mu}{1+\mu})\sin\theta - \Omega\frac{\beta\gamma\mu\delta}{(1+\mu)}\sin(\theta+\phi) + \rho_2\ddot\phi +$$

$$(\frac{\gamma\mu\delta^2\beta}{1+\mu}\sin\phi(2\dot\theta+\dot\phi)\dot\phi + (\frac{\gamma^2\mu\delta^2\beta^2}{1+\mu} - \frac{\gamma\mu\delta^2\beta}{1+\mu}\cos\phi)\ddot\phi = 0, \quad \text{(B.24)}$$

Bibliography

E. W. Aboaf, S. M. Drucker, and C. G. Atkeson. Task-level robot learning: Juggling a tennis ball more accurately. In *IEEE International Conference on Robotics and Automation*, pages 1290–1295, Scottsdale, AZ, 1989.

K. Autumn, M. Buehler, M. Cutkosky, R. Fearing, R. J. Full, D. Goldman, R. Groff, W. Provancher, A. A. Rizzi, U. Saranli, A. Saunders, and D. E. Koditschek. Robotics in scansorial environments. In *Proc. of SPIE Vol. 5804 Unmanned Ground Vehicle Technology VII*, pages 291–302, 2005.

M. D. Berkemeier and R. S. Fearing. Sliding and hopping gaits for the underactuated acrobot. *IEEE Transactions on Robotics and Automation*, 14(4):629–634, August 1998.

D. Bevly, S. Dubowsky, and C. Mavroidis. A simplified cartesian-computed torque controller for highly geared systems and its application to an experimental climbing robot. *Journal of Dynamic Systems, Measurement, and Control*, 122:27, 2000.

R. Blickhan. The spring-mass model for running and hopping. *Journal of Biomechanics*, 22: 1217–1227, 1989.

R. Blickhan and R. J. Full. Similarity in multilegged locomotion: Bouncing like a monopode. *Journal of Comparative Physiology*, 173:509–517, 1993.

W. E. Boyce and R. C. DiPrima. *Elementary Differential Equations and Boundary Value Prob-*

lems. John Wiley & Sons, 2001.

T. Bretl. Motion planning of multi-limbed robots subject to equilibrium constraints: The free-climbing robot problem. *The International Journal of Robotics Research*, 25(4):317–342, 2006.

H. B. Brown, Jr. and G. Zeglin. The Bow Leg hopping robot. In *IEEE International Conference on Robotics and Automation*, pages 781–786, 1998.

H. B. Brown, Jr., G. Zeglin, and I. R. Nourbakhsh. Resilient leg design for hopping running and walking machines. Patent, September 2007. US 7,270,589 B1.

M. Bühler and D. E. Koditschek. From stable to chaotic juggling: Theory, simulation, and experiments. In *IEEE International Conference on Robotics and Automation*, pages 1976–1981, Cincinnati, OH, 1990.

R. R. Burridge, A. A. Rizzi, and D. E. Koditschek. Sequential composition of dynamically dexterous robot behaviors. *The International Journal of Robotics Research*, 18(6):534–555, jun 1999.

K. Byl and R. Tedrake. Metastable walking machines. *The International Journal of Robotics Research*, 28(8):1040, 2009.

J.F. Canny and K.V. Goldberg. A RISC approach to sensing and manipulation. *Journal of Robotic Systems*, 12(6), 1995.

J. E. Clark, D. I. Goldman, T. S. Chen, R. J. Full, and D. Koditschek. Toward a dynamic climbing robot. In *Proc. of the 9th International Conference on Climbing and Walking Robots (CLAWAR '06)*, Brussels, Belguim, September 2006.

J.E. Clark, D.I. Goldman, P.C. Lin, and G. Lynch. Design of a Bio-inspired Dynamical Vertical Climbing Robot. *Robotics: Science and Systems III Atlanta, Georgia*, 2007.

S. Collins, A. Ruina, R. Tedrake, and M. Wisse. Efficient bipedal robots based on passive-

dynamic walkers. *Science*, 307(5712):1082–1085, February 2005.

S. H. Collins, M. Wisse, and A. Ruina. A three-dimensional passive-dynamic walking robot with two legs and knees. *International Journal of Robotics Research*, 20(7):607–615, July 2001a.

S. H. Collins, M. Wisse, and A. Ruina. A three-dimensional passive-dynamic walking robot with two legs and knees. *The International Journal of Robotics Research*, 20(7):607–615, 2001b.

A. Degani, A. Shapiro, H. Choset, and M. T. Mason. A dynamic single actuator vertical climbing robot. In *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'07)*, 2007.

A. Degani, H. Choset, and M. T. Mason. DSAC – Dynamic, Single Actuated Climber: Local stability and bifurcations. In *Proc. of the IEEE International Conference on Robotics and Automation, (ICRA)*, 2010a.

A. Degani, H. Choset, and M. T. Mason. Minimalistic, dynamic, tube climbing robot - video submission. In *Proc. of the IEEE International Conference on Robotics and Automation, (ICRA)*, 2010b.

A. Degani, H. Choset, and M. T. Mason. Minimalism in climbing locomotion: A single actuator climbing. *IEEE Transactions on Robotics - under review*, 2010c.

A. Degani, H. Choset, and M. T. Mason. DTAR – a dynamic tube ascending robot. *IEEE Transactions on Robotics - under review*, 2010d.

A. Degani, S. Feng, H. B. Brown, K. M. Lynch, H. Choset, and M. T. Mason. The ParkourBot – a dynamic bowleg climbing robot. In *Proc. of the IEEE International Conference on Robotics and Automation, (ICRA) - under review*, May 2011.

H.R. Dullin. Melnikov's method applied to the double pendulum. *Zeitschrift fur Physik B Condensed Matter*, 93(4):521–528, 1994.

M. A. Erdmann and M. T. Mason. An exploration of sensorless manipulation. *IEEE Transactions*

on Robotics and Automation, 4(4):369–379, August 1988.

M. Garcia, A. Chatterjee, A. Ruina, and M. Coleman. The simplest walking model: stability, complexity, and scaling. *Journal of Biomechanical Engineering*, 120(2):281–288, April 1998.

H. Geyer, R. Blickhan, and A. Seyfarth. Natural dynamics of spring-like running: Emergence of selfstability. In *Proceedings of the Fifth International Conference on Climbing and Walking Robots: and their supporting technologies: CLAWAR 2002, 25-27th September 2002*, page 87. Wiley, 2002.

H. Geyer, A. Seyfarth, and R. Blickhan. Spring-mass running: simple approximate solution and application to gait stability. *Journal of theoretical biology*, 232(3):315–328, 2005.

RM Ghigliazza, R. Altendorfer, P. Holmes, and D. Koditschek. A simply stabilized running model. *SIAM review*, 47(3):519, 2005.

PR Gill, W. Murray, and MH Wright. The levenberg-marquardt method. *Practical Optimization*, pages 136–137, 1981.

A. Gmiterko, M. Dovica, M. Kelemen, V. Fedak, and Z. Mlynkova. In-pipe bristled microma-chine. In *7th International Workshop on Advanced Motion Control*, 2002.

A. Goswami, B. Thuilot, and B. Espiau. A study of the passive gait of a compass-like biped robot: symmetry and chaos. *The International Journal of Robotics Research*, 17(12):1282, 1998.

A. Greenfield, A. A. Rizzi, and H. Choset. Dynamic ambiguities in frictional rigid-body systems with application to climbing via bracing. In *Proc. of the IEEE International Conference on Robotics and Automation, (ICRA)*, pages 1947– 1952, Barcelona, Spain, 2005.

D. T. Greenwood. *Classical dynamics*. Dover Publications, Mineola, N.Y., 1997.

R.D. Gregg, T. Bretl, and M.W. Spong. Asymptotically stable gait primitives for planning dy-namic bipedal locomotion in three dimensions. In *Robotics and Automation (ICRA), 2010*

IEEE International Conference on, pages 1695–1702. IEEE, 2010.

J. Guckenheimer and P. Holmes. *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*. Springer, 1983.

D.G.E. Hobbelen and M. Wisse. A disturbance rejection measure for limit cycle walkers: The gait sensitivity norm. *Robotics, IEEE Transactions on*, 23(6):1213–1224, 2007.

J. K. Hodgins and M. H. Raibert. Biped gymnastics. *International Journal of Robotics Research*, 9(2):115–132, 1990.

C.S Hsu. *Cell-to-cell mapping: a method of global analysis for nonlinear systems*. Springer, New York, 1997.

S. Jensen-Segal, S. Virost, and WR Provancher. ROCR: Dynamic vertical wall climbing with a pendular two-link mass-shifting robot. In *IEEE International Conference on Robotics and Automation, (ICRA)*, pages 3040–3045, 2008.

I. Kassim, L. Phee, WS Ng, F. Gong, P. Dario, and CA Mosse. Locomotion techniques for robotic colonoscopy. *IEEE Engineering in Medicine and Biology Magazine*, 25(3):49–56, 2006.

I. Kato, S. Ohteru, H. Kobayashi, K. Shirai, and A. Uchiyama. Information-power machine with senses and limbs. In *First CISM-IFToMM Symposium on Theory and Practice of Robots and Manipulators*. Springer-Verlag, 1974.

S. Kim, A. T. Asbeck, M. R. Cutkosky, and W. R. Provancher. SpinybotII: Climbing hard walls with compliant microspines. In *Proc. of the 12th International Conference on Advanced Robotics, (ICAR '05)*, pages 601–606, Seattle, WA, 2005.

S. Kim, M. Spenko, S. Trujillo, B. Heyneman, V. Mattoli, and M. R. Cutkosky. Whole body adhesion: hierarchical, directional and distributed control of adhesive forces for a climbing robot. In *IEEE International Conference on Robotics and Automation*, pages 1268–1273, 2007.

179

D. E. Koditschek and M. Bühler. Analysis of a simplified hopping robot. *International Journal of Robotics Research*, 10(6):587–605, December 1991.

A. D. Kuo. Energetics of actively powered locomotion using the simplest walking model. *Journal of Biomechanical Engineering*, 124(1):113–120, February 2002.

D. Longo and G. Muscato. The Alicia[3] climbing robot. *IEEE Robotics and Automation Magazine*, 13:2–10, 2006.

T. Lozano-Perez, M.T. Mason, and R.H. Taylor. Automatic synthesis of fine-motion strategies for robots. *The International Journal of Robotics Research*, 3(1):3, 1984.

K. M. Lynch and Craig K. Black. Recurrence, controllability, and stabilization of juggling. *IEEE Transactions on Robotics and Automation*, 17(2):113–124, April 2001.

K.M. Lynch and M.T. Mason. Dynamic nonprehensile manipulation: Controllability, planning, and experiments. *International Journal of Robotics Research*, 1999.

M.T. Mason. The mechanics of manipulation. In *Proceedings of the 1985 IEEE International Conference on Robotics and Automation*, pages 544–548, 1985.

M.T. Mason. *Mechanics of Robotic Manipulation*. The MIT Press, 2001.

T. McGeer. Passive walking with knees. In *IEEE International Conference on Robotics and Automation*, pages 1640–1645, 1990a.

T. McGeer. Passive dynamic walking. *The International Journal of Robotics Research*, 9(2): 62–82, April 1990b.

A. Menciassi, D. Accoto, S. Gorini, and P. Dario. Development of a biomimetic miniature robotic crawler. *Autonomous Robots*, 21(2):155–163, 2006.

M. Milano, D. Jurjevich, and R. Adrian. The flying brick: A cautionary note on testing flying robots using guide wires. *IEEE Transactions on Robotics*, 25(2):426–428, April 2009.

V. Mištinas and B. Spruogis. Development of pipe crawling robots with vibratory drives and investigation of their kinematic parameters. *Transport*, 17:171–176, 2002.

K. D. Mombaur, H. G. Bock, J. P. Schloder, and R. W. Longman. Self-stabilizing somersaults. *IEEE Transactions on Robotics*, 21(6):1148–1157, December 2005a.

K.D. Mombaur, R.W. Longman, H.G. Bock, and J.P. Schlöder. Open-loop stable running. *Robotica*, 23(01):21–33, 2005b.

M. Murphy and M. Sitti. Waalbot: An agile small-scale wall climbing robot utilizing pressure sensitive adhesives. *IEEE/ASME Trans. on Mechatronics*, 12(3):330–338, 2007.

R. M. Murray, Z. Li, and S. S. Sastry. *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Boca Raton, FA, 1994.

NaturalPoint^TM. Optitrack camera system. Retrieved from www.naturalpoint.com/optitrack, July 2009.

A.H. Nayfeh and B. Balachandran. *Applied nonlinear dynamics*. Wiley New York, 1995.

T.S. Parker and L.O. Chua. *Practical numerical algorithms for chaotic systems*. Springer-Verlag New York, Inc. New York, NY, USA, 1989.

M. H. Raibert. *Legged Robots That Balance*. The MIT Press, Cambridge, MA, 1986.

M. H. Raibert and H. B. Brown, Jr. Experiments in balance with a 2D one-legged hopping machine. *ASME Journal of Dynamic Systems, Measurement, and Control*, 106:75–81, 1984.

M. H. Raibert, H. B. Brown, Jr., and M. Chepponis. Experiments in balance with a 3D one-legged hopping machine. *International Journal of Robotics Research*, 3:75–92, 1984.

M. H. Raibert, M. Chepponis, and H. B. Brown, Jr. Running on four legs as though they were one. *IEEE Journal on Robotics and Automation*, 2:70–82, 1986.

R. Ringrose. Self-stabilizing running. In *IEEE International Conference on Robotics and Au-*

tomation, pages 487–493, 1997.

A. A. Rizzi and D. E. Koditschek. Progress in spatial robot juggling. In *IEEE International Conference on Robotics and Automation*, pages 775–780, Nice, France, 1992.

R. Ronsse, P. Lefevre, and R. Sepulchre. Sensorless stabilization of bounce juggling. *IEEE Transactions on Robotics*, 22(1):147–159, February 2006.

O. Salomon, N. Shvalb, and M. Shoham. Vibrating robotic crawler. Patent WO/2008/12608, apr 2008.

S. Schaal and C. G. Atkeson. Open loop stable control strategies for robot juggling. In *Proc. of the IEEE International Conference on Robotics and Automation, (ICRA)*, pages 913–918, Atlanta, GA, 1993.

A. Seyfarth, H. Geyer, M. G
"unther, and R. Blickhan. A movement criterion for running. *Journal of Biomechanics*, 35(5): 649, 2002.

A. Seyfarth, H. Geyer, and H. Herr. Swing-leg retraction: a simple control model for stable running. *Journal of Experimental Biology*, 206:2547–2555, 2003.

A. Shapiro, E. Rimon, and S. Shoval. PCG: A foothold selection algorithm for spider robot locomotion in 2D tunnels. *The International Journal of Robotics Research*, 24(10):823–844, 2005.

AB Slatkin and J. Burdick. The development of a robotic endoscope. In *Proc. of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'95)*, page 2162, 1995.

J.C. Sprott. *Chaos and Time-Series Analysis*. Oxford University Press, 2003.

DE Stewart and JC Trinkle. An implicit time-stepping scheme for rigid body dynamics with inelastic collisions and coulomb friction. *International Journal for Numerical Methods in Engineering*, 39(15), 1996.

S.H. Strogatz. Non Linear Dynamics. *Adison Wesley*, 1994.

G. Takanishi, M. Ishida M. Naito, and I. Kato. Realization of plane walking by the biped walking robot WL-10R. *Robotic and Manipulator Systems*, pages 283–393, 1982.

R. Tedrake, I.R. Manchester, M. Tobenkin, and J.W. Roberts. LQR-trees: Feedback Motion Planning via Sums-of-Squares Verification. *The International Journal of Robotics Research*, 2010.

A. F. Vakakis, J. W. Burdick, and T. K. Caughey. An interesting strange attractor in the dynamics of a hopping robot. *International Journal of Robotics Research*, 10(6):606–618, December 1991.

M. Vukobratovic and B. Borovac. Zero-moment point-thirty five years of its life. *International Journal of Humanoid Robotics*, 1(1):157–173, 2004.

M. Vukobratovic and D. Juricic. A Contribution to the Synthesis of Biped Gait. In *IFAC Symp. Technical and Biological Problem on Control*, Erevan, USSR, 1968.

K. Wang, G. Yan, P. Jiang, and D. Ye. A wireless robotic endoscope for gastrointestine. *IEEE Transactions on Robotics*, 24(1):206–210, 2008.

J. Weingarten, G. Lopes, M. Buehler, R. Groff, and D. Koditschek. Automated gait adaptation for legged robots. In *IEEE International Conference on Robotics and Automation*, volume 3, pages 2153–2158. IEEE; 1999, 2004.

G. Zeglin and H. B. Brown, Jr. Control of a Bow Leg hopping robot. In *IEEE International Conference on Robotics and Automation*, pages 793–798, 1998.

Garth Zeglin. *The Bow Leg Hopping Robot*. PhD thesis, Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, October 1999.