

ON PERFORMANCE AND STABILITY IN OPEN-  
LOOP RUNNING

A DISSERTATION  
SUBMITTED TO THE DEPARTMENT OF MECHANICAL ENGINEERING  
AND THE COMMITTEE ON GRADUATE STUDIES  
OF STANFORD UNIVERSITY  
IN PARTIAL FULFILLMENT OF THE REQUIREMENTS  
FOR THE DEGREE OF  
DOCTOR OF PHILOSOPHY

Jorge Gabriel Cham  
December 2002

© Copyright 2003 by Jorge Cham  
All Rights Reserved

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

---

Mark R. Cutkosky Principal Advisor

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

---

Kenneth J. Waldron

I certify that I have read this dissertation and that in my opinion it is fully adequate, in scope and quality, as a dissertation for the degree of Doctor of Philosophy.

---

J. Christian Gerdes

Approved for the University Committee on Graduate Studies:

---

# Abstract

Legged robots can provide access to hazardous environments such as waste sites, disaster areas and even the surfaces of other planets, where large obstacles and uneven terrain impede mobility with traditional wheeled systems. Looking to nature for design inspiration, a family of hexapedal robots has been developed that rely on passive mechanical properties and a simple open-loop motor control program to achieve fast and robust performance that begins to compare to that seen in nature. These robots are capable of running at over 4 body-lengths/second and can overcome hip-height obstacles without significantly slowing down. Although this approach works well for a range of open-loop parameters, the question of how to "tune" the system to achieve maximum performance while maintaining stability, especially as conditions change, arises. This thesis analyzes the trade-offs in stability and performance for open-loop running systems like the robots starting with simplified one and two degree-of-freedom dynamic models. Analyses of these models are then validated with experimental data of the running hexapods. Results reveal conditions for optimal performance and demonstrate inherent limitations in the use of open-loop control in running. Based on these findings, strategies for adaptation, or self-tuning, of the open-loop motor program are proposed that require only simple, low-bandwidth sensory feedback. One of these strategies is demonstrated experimentally in the adaptation of stride period in a hexapedal running robot.

*For my real-life superheroes:  
my parents, Winston and Cecilia*

# Acknowledgments



JORGE CHAM © 2002

# Contents

<b>CHAPTER 1. Introduction.....</b>	<b>1</b>
1.1 Context and Motivation	1
1.2 Thesis Overview	6
1.3 Contributions	8
<b>CHAPTER 2. Approach .....</b>	<b>10</b>
2.1 Analysis of Locomotion via Poincare Maps	11
2.1.1 Poincare Maps and Poincare Sections	12
2.1.2 Poincare Maps for Clock-driven Systems	13
2.1.3 Steady-state Trajectories and Fixed Points	15
2.1.4 Poincare Maps for Hybrid Systems	16
2.2 Resonance in Running	19
2.3 Stability of Steady-state Trajectories	21
2.3.1 Local Stability and Floquet Multipliers	21
2.3.2 Derivation of Jacobian for Hybrid Systems	23
2.3.3 Jacobian Similarity Transformation	25
2.4 Chapter Summary	26
<b>CHAPTER 3. Open-loop Vertical Hopping.....</b>	<b>28</b>
3.1 Related Work	28
3.2 Vertical Hopper Model	30
3.2.1 Model Description	31
3.2.2 Derivation of Poincare Map for Steady-state Solutions	33
3.2.3 Derivation of Jacobians for Local Stability	36
3.2.4 Solution Search Results	38
3.3 Characterizing Performance	42
3.4 Characterizing Stability	45
3.4.1 Stability for "Long Thrust" Case	46
3.4.2 Stability for "Short Thrust" case	51
3.5 Experimental Verification - Dashpod Data	55
3.6 Chapter Conclusions	58
<b>CHAPTER 4. Open Loop Planar Running.....</b>	<b>61</b>
4.1 Model Description	61
4.1.1 Equations of Motion	63
4.1.2 Poincare Map and Jacobian	64
4.2 Basic SLIP Running	66
4.2.1 Performance in Basic SLIP Running	67

4.2.2 Stability of Basic SLIP	71
4.2.3 Alternate Stabilization of the Basic SLIP	73
4.3 Sprawl-type Running	76
4.3.1 Performance and Stability	78
4.4 Chapter Conclusions	84
<b>CHAPTER 5. Stride Period Adaptation for a Hexapedal Running Robot.....</b>	<b>87</b>
5.1 Stride Period Adaptation	87
5.2 Robot Performance Tests	89
5.3 Adaptation Strategy	93
5.4 Adaptation Results	95
5.5 Chapter Conclusions	98
<b>CHAPTER 6. Conclusions.....</b>	<b>100</b>
6.1 Main Conclusions	100
6.2 Suggestions for Future Work	102
<b>Appendix .....</b>	<b>104</b>
<b>Bibliography .....</b>	<b>115</b>

# Figures

<b>CHAPTER 1.</b>	1
Figure 1-1. A combination of stabilizing passive mechanisms, or "preflexes," and sensor-based adaptation of an open-loop feed-forward controller provides insects and small robots with a robust, stable and versatile approach to running over rough terrain. Adapted from (Full and Koditschek, 2000).....	2
Figure 1-2. A family of hexapedal robots has been developed that demonstrates that fast and robust locomotion is possible without feedback control. The body and legs were fabricated by Shape Deposition Manufacturing (Cham et al., in press) and feature embedded actuators and compliant legs. On the upper right, the robot is shown crossing a hip-height obstacle without using sensory feedback and without significantly slowing down or being knocked off course.....	4
Figure 1-3. Robot ground speed versus terrain slope for two different stride frequencies of the open-loop motor pattern. As shown, stable running occurs over a range of frequencies, although the stride frequency for maximum speed depends on the slope, which illustrates the need to understand how performance and stability change as a function of the motor pattern. ....	5
<b>CHAPTER 2.</b>	10
Figure 2-1. The dynamics of open-loop running are studied through analysis of simplified vertical and planar models and experimental data of the Sprawl Robots. ....	11
Figure 2-2. Graphical illustration of a Poincare Map and Poincare Section.....	12
Figure 2-3. Graphical illustration of the addition of a proxy "clock" state variable for non-autonomous systems driven by a repeating motor pattern. .	15
Figure 2-4. Graphical illustration of modes and events in a general hopping system. Each node in the graph represents a mode of the system, governed by its corresponding equations of motion. The arrows represent events in the state-space or in time that cause the system to transition to another mode. ....	17
Figure 2-5. Graphical illustration of the Jacobian, which maps perturbations about a steady-state trajectory, $x_0(t)$ , between subsequent intersections with the Poincare Section. ....	22
Figure 2-6. Graphical illustration of sensitivity matrices. Along a given steady-state trajectory (shown dotted), there may exist multiple mode transitions. The Jacobian for the Poincare Map can be written as the	

product of a set of "sensitivity" matrices, which are the multivariable derivatives of the maps  $M_i$  between the states at the mode transitions, given by the intersections with the corresponding mode boundaries. 24

Figure 2-7. Graphical illustration of two Poincare Maps for a given steady-state trajectory. The Jacobians for both Poincare Maps are related by a similarity transformation. .... 25

**CHAPTER 3.** ..... 28

Figure 3-1. The vertical hopping model used for analysis. The hopper's leg consists of a spring, a damper and a force element which is active according to a binary motor pattern. The figure shows a sample trajectory of the hopper, the different modes that it goes through, and the events that trigger the transitions between the modes. .... 31

Figure 3-2. Illustration of a sample time history of the vertical hopper. The figure shows the two possibilities for formulating the Poincare Map used in analysis: a Map based on the state at thrust activation, and a Map based on the velocity and time at landing. .... 33

Figure 3-3. Sample time history the hopper with the reduced states shown at each mode boundary. .... 36

Figure 3-4. Steady-state solutions as a function of stride period for the "Long Thrust" case. The arrows indicate how the continuum of solutions folds unto itself. The arrows also show how the continuum of the magnitude of the eigenvalues splits from two complex eigenvalues with the same magnitude to an unstable and a stable real eigenvalue... 38

Figure 3-5. Multiple steady-state solutions for a given stride period. The plot in the lower right shows the continuum of steady-state solutions plotted in the state-space of the state at thrust activation. The continuum is separated into three segments, each characterized by when thrust is initiated relative to maximum compression. .... 39

Figure 3-6. Steady-state solutions as a function of stride period for the "Short Thrust" case. Arrows indicate how the continuum of solutions folds unto itself. At some point, thrust deactivation starts to occur after take-off, and the continuum switches to "Long Thrust" solutions. .... 41

Figure 3-7. Height and work versus toff, the time after landing that thrust is initiated. As shown, solutions in which maximum work is performed correspond to maximum height. Note also that multiple solutions do not exist when plotting against toff. .... 43

Figure 3-8. Work performed by the actuator in this model is proportional to the height difference between the state at thrust activation and thrust deactivation. 44

Figure 3-9. Unstable steady-state solution for "Long Thrust" case. At a period of 0.26, there are three valid steady-state solutions. The solution in which thrust is activated shortly after maximum compression is unstable, as

	illustrated by the vector field. In the vector field, the dots indicate the state at the time that thrust is initiated. The vector lines indicate the direction from this initial condition towards the state after one period of the motor pattern. ....	45
Figure 3-10.	Effect of perturbations in the timing of thrust activation on the work performed, and resulting take-off velocity and airborne phase duration for the "Long Thrust" case.....	49
Figure 3-11.	Effect of perturbations in the timing of thrust activation on the work performed, and resulting take-off velocity and airborne phase duration for the "Short Thrust" case. ....	50
Figure 3-12.	Unstable steady-state solution for "Short Thrust" case for a period of 0.15. At this period, thrust is initiated well before maximum compression, and take-off velocity is small. ....	51
Figure 3-13.	Stride periods for which unstable period-1 trajectories exist in the short thrust case tend to result in period-2 stable trajectories. These period-2 orbits alternately initiate thrust before and after the nominal point in the cycle where the unstable period-1 orbit initiates thrust. This behavior is explained by the negative sign of the eigenvalues in this regime, which indicate that perturbations about the nominal trajectory will be "overcompensated", causing thrust activation to oscillate about the unstable equilibrium. ....	54
Figure 3-14.	The "Dashpod," a pneumatically-actuated vertical hopper. ....	55
Figure 3-15.	Experimental data of hopping trajectories for the Dashpod vertical hopper. These nine trials represent variations in the hopper's mass and the period of the motor pattern. ....	57
<b>CHAPTER 4.</b>	.....	61
Figure 4-1.	General planar running model. A point mass is attached to a massless leg with radial passive and active elements in parallel, and a "foot" and "ankle" which provide rotational stiffness and damping when in contact with the ground. ....	62
Figure 4-2.	Poincare Map between landing events. The reduced state vector mapped consists of the velocity of the mass (magnitude and direction), and toff, the time between landing and thrust activation.....	64
Figure 4-3.	Illustration of "Basic SLIP" and "Sprawl-type" variations on the planar running model. The Basic SLIP has no ankle compliant element, and is characterized by trajectories in which the leg is placed forward for landing. In the "Sprawl-type" running model, a compliant ankle stabilizes the direction of the leg, and the leg is placed back for landing.	66
Figure 4-4.	Steady-state solutions for Basic SLIP model with varying leg landing angle. Note that for open-loop control, there exist multiple solutions for a range of stride periods.....	68

Figure 4-5.	Steady-state solutions for Basic SLIP plotted against toff. Solutions in the mid-ranges of toff represent trajectories in which thrust is initiated near maximum compression and have longer stride lengths and hopping height, but are suboptimal in terms of forward speed. Solutions in which thrust is initiated shortly after landing, with minimal toff, have higher forward velocities. ....	69
Figure 4-6.	Eigenvalues of the Jacobian for the case of open-loop control of the SLIP. As shown, "skittering" hopping trajectories are less unstable, and instability is dominated by one mode. ....	71
Figure 4-7.	Magnitudes of the elements of the eigenvector corresponding to the largest and median eigenvalues. As shown, the unstable mode with the largest eigenvalue is primarily in the direction of disturbances in the angle of the landing velocity. The unstable mode that appears for longer values of toff is primarily in the direction of disturbances in toff.	72
Figure 4-8.	a) Unstable mode in the direction of the landing velocity for the Basic SLIP. Disturbances in the direction of the landing velocity result in large perturbations in the direction of the take-off velocity. b) Effect of disturbances in toff, the time of thrust activation. It can be shown that toff can be used in a control law that stabilizes a hopper with open-loop leg landing angle. ....	73
Figure 4-9.	Inverse of the condition number for the controllability matrix. Non-zero values indicate that the system is controllable, and that the basic SLIP can be stabilized by controlling the thrust timing toff. As shown, for mid-range values of toff, which correspond to initiating thrust near maximum compression, the effect of toff on the magnitude and direction of the landing velocity decreases, and toff loses "control authority."	75
Figure 4-10.	Captured images of Sprawlita in locomotion. As shown, the hip flexures play a large role in locomotion, and undergo significant deflection during stance. ....	76
Figure 4-11.	Forward speed of the steady-state solutions for a range of ankle stiffnesses and leg landing angles. As shown, speed is maximized at the borderline of the region for which solutions exist. ....	77
Figure 4-12.	Effect of increasing the leg landing angle. By increasing the landing angle, the equilibrium position of the ankle increases, due to the increased effect of gravity. If the equilibrium position is too steep, the system cannot recover and no solution will exist. Decreasing the ankle stiffness has a similar effect. ....	78
Figure 4-13.	Stride length of the steady-state solutions for a range of ankle stiffnesses and leg landing angles. As shown, stride length is not maximized at the same values of toff as forward speed. ....	79
Figure 4-14.	Work performed by the actuator in the steady-state solutions for a range of ankle stiffnesses and leg landing angles. As shown,	

combinations of ankle stiffness and leg landing angle with maximum peak work do not correspond to combinations with maximum peak stride length. However, for each particular combination and ankle stiffness and landing angle, the optimal value of toff which maximizes stride length is near the same toff which maximizes work. .... 80

Figure 4-15. Correlation between maximum work and maximum stride length. The stride period for which stride length is maximized over the continuum of solutions for a given combination of landing angle and ankle stiffness is plotted against the corresponding stride period for which work is maximized. Each line represents the period for a given leg landing angle. .... 81

Figure 4-16. Magnitude of the eigenvalues of the Jacobian of the Poincare Map for the continuum of steady-state solutions, given a range of leg landing angles and ankle stiffnesses. As shown, the maximum magnitudes increase near low values of toff, and near mid-range values of toff, in which thrust is activated after maximum compression. .... 82

**CHAPTER 5.** ..... 87

Figure 5-1. Adaptation scheme in which rapid disturbances are rejected by the passive stabilizing properties of the mechanical system. Sensory information is used at a slower rate to adapt the parameters of the open-loop motor pattern in response to changes in the environment or loading conditions. .... 88

Figure 5-2. Experimental results of performance tests in Sprawlita, one of the Sprawl robots. The dotted line represents results for the robot with pneumatic actuators with significant damping. Solid lines represent results for the robot with pistons with less damping. Dashed lines represent the second robot running on a 5 degree sloped surface. As shown, despite the variation in robot parameters, the stride period for which stride length is maximized is correlated to a change in slope in the time difference between thrust deactivation and take-off. .... 90

Figure 5-3. Illustration of indirectly measuring work performed by the actuator by measuring the time difference between take-off and actuator deactivation. Optimal work is done when the thrust is deactivated just as the leg reaches maximum extension. As shown, when thrust is deactivated before maximum extension, the time difference is a low constant value, as take-off occurs due to leg retraction. When thrust is deactivated well after maximum extension, the time difference is an increasing positive value. .... 92

Figure 5-4. Stride period adaptation strategy. The stride period is updated after every stride cycle based on the measured times that valve deactivation and loss of contact occur. As illustrated, the adaptation strategy seeks to find the period in which work is maximized, given by the period in which the time difference changes slope, and exceeds the threshold value  $t_v$ . .... 94

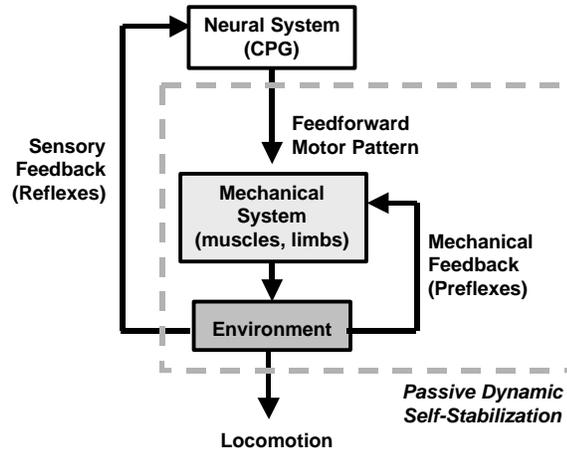
Figure 5-5.	Adaptation results for flat terrain (dashed lines are approximate values for maximum speed established empirically). The figures show the ground speed of the robot and the stride period as it is adapted from suboptimal starting conditions.....	95
Figure 5-6.	Adaptation results for flat terrain with a 13% decrease in pneumatic actuator input pressure. The adaptation optimizes ground speed by converging to a slightly higher stride period than for the lower input pressure.	96
Figure 5-7.	Adaptation results for an uphill slope of 5 degrees. The adaptation strategy improves the locomotion, but converges to a stride period slightly higher than the optimal stride period.....	97
<b>CHAPTER 6.</b>	.....	100

# 1 Introduction

## 1.1 Context and Motivation

Legged robots can provide access to unstructured environments such as waste sites, disaster areas and even the surfaces of other planets. While it is currently difficult to compete with the simplicity and established performance of wheeled systems, one need only look at the pervasiveness of legged animals in the natural world and the successful locomotion of even the simplest of insects to assert that legged systems can be more versatile and more adept at negotiating uneven terrain and large obstacles than equivalent wheeled vehicles. As a result of this observation, legged robot research has looked at biological systems for design inspiration (Ritzman et al., 2000). The most common instance of this "biomimicry" is seen in the large number of walking robots that utilize six legs in a variety of gaits intended to maintain static stability (for example, Bares et al., 1999 and Waldron, 1986). However, unlike animals, few legged robots have exhibited the robustness and versatility needed to operate with significant speed in unstructured environments, despite advances in control and artificial intelligence.

One reason for the limited performance of legged robots has been the use of design, fabrication and control methods traditionally used in robotics for applications in structured environments like laboratories or factory floors. These methods rely on assemblies of stiff metal structures, bearings and fasteners that are controlled using high-bandwidth sensors and actuators, and most commonly utilize some form of impedance control (Hogan, 1985) for disturbance rejection. Under impedance control, the robot's actuators impart forces on the robot's stiff limbs in response to sensed disturbances in order to achieve desired inertia, stiffness and damping properties. Although this approach works well for slow operation in structured environments, unexpected link collisions can exceed closed-loop bandwidth limitations, resulting in high transient forces that can damage the system.



**Figure 1-1.** A combination of stabilizing passive mechanisms, or "preflexes," and sensor-based adaptation of an open-loop feed-forward controller provides insects and small robots with a robust, stable and versatile approach to running over rough terrain. Adapted from (Full and Koditschek, 2000).

In contrast, nature's mechanisms are robust. Sensors, actuators and structural elements are compactly integrated and enclosed in structures that protect them against harsh external conditions and avoid stress concentrations that cause failure. Moreover, these structures are constructed using soft materials frequently and stiff materials sparingly (Vogel, 1995), allowing them to tolerate greater overloads.

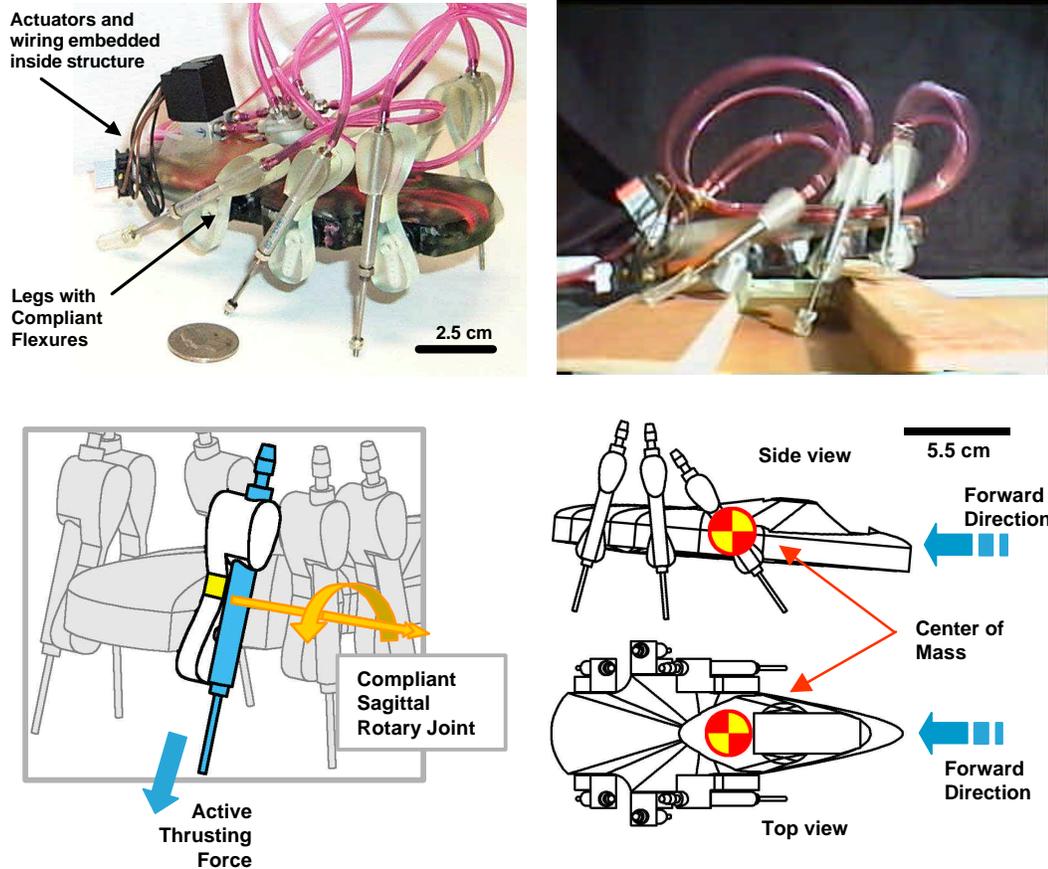
The use of compliant materials not only increases sturdiness against sudden impacts, but also provides advantages in control. Appropriately located compliance and damping can improve performance and simplify control by providing passive mechanisms for energy storage and return and for stabilization and disturbance rejection. These passive visco-elastic properties are at the heart of robust locomotion (McMahon, 1984), and have been termed "preflexes" (van Soest and Bobbert, 1993; Brown and Loeb, 1997).

Studies of the control of dynamic rhythmic movements like running and walking in animals confirm that control is not predominantly based on sensory feedback (Abbas and Full, 2001; Full and Koditschek, 2000). The presence of neural circuits called Central Pattern Generators (CPG) that can generate, in the absence of feedback, efferent motor patterns similar to those seen during locomotion is well established in both vertebrates and invertebrates. These efferent motor patterns form the basis of limb movements during locomotion, similar to a feedforward control signal. For example, studies of the cockroach running over uneven terrain show that there are only minor changes in the cockroach's muscle

activation pattern as it rapidly transitions from smooth to uneven terrain (Full et al., 1998). There are no carefully controlled foot placements or noticeable changes in gait pattern.

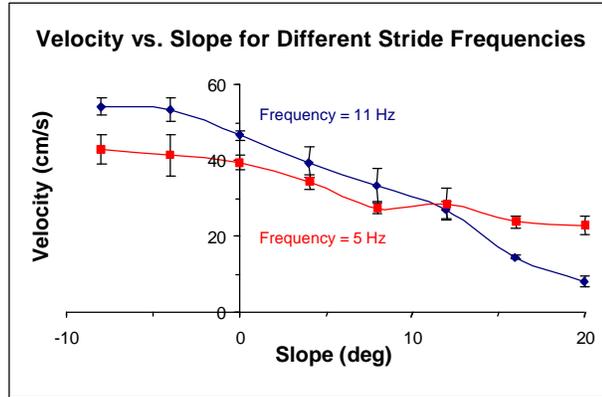
Although the pattern generators can function without feedback, signals from receptors in the skin, joints and muscles play an important role in modulating the output frequency and amplitude (Rossignol, et al., 1988; Orlovsky, et al., 1999; Abbas and Full, 2001). The picture that emerges is illustrated by Figure 1-1. In this hypothesized control architecture, an open-loop, feed-forward, motor controller generates the pattern of actuator commands for a steady running gait. The kinematic arrangement and passive compliance and damping of the limbs achieve the locomotion and provide immediate response to perturbations. Sensory information is used at a slower rate to adapt, or tune, the motor pattern in response to changing conditions. In running animals, an important reason for relying on reflexes in combination with slow adaptation is that neural conduction speeds are often too slow for feedback control to act effectively within each stride, especially at high speeds. In robots, of course, the same limitation does not necessarily apply. However, for robots, the same approach allows the use of simple sensors without concerns that actuator delays, sensor noise or even failures will jeopardize short-term performance.

Several researchers have shown that dynamic robotic locomotion is possible with little or no active control. McGeer (1990) showed that a biped, constructed of properly sized stiff links, can walk passively down an incline without feedback control. The pioneering work of Raibert (1986) showed that a monopod can hop and run taking advantage of passive compliance and that its control can be decoupled into a set of simple laws for the control of height, speed and attitude. More recently, it has been shown that hopping and running can be achieved without feedback control. Ringrose (1998) showed that a vertical hopper with a curved foot can be actuated by a pre-programmed, repeating motor control pattern and still be "self-stabilizing" in terms of hopping height and phasing with respect to the actuator pattern. These vertical hoppers were combined in pairs in simulation to form stable bipeds and quadrupeds. Mombaur (2001) showed in simulation that it is possible to find stable open-loop running trajectories for a multi-body, multiple degree-of-freedom (DOF) planar hopper with a pointed foot.



**Figure 1-2.** A family of hexapedal robots has been developed that demonstrates that fast and robust locomotion is possible without feedback control. The body and legs were fabricated by Shape Deposition Manufacturing (Cham et al., in press) and feature embedded actuators and compliant legs. On the upper right, the robot is shown crossing a hip-height obstacle without using sensory feedback and without significantly slowing down or being knocked off course.

Also recently, a family of hexapedal robots developed in the work leading to this thesis shows that fast and robust performance can be achieved without closed-loop control (Cham et al., 1999; Cham et al., in press). Inspired by studies of the cockroach, these hexapedal prototypes are able to run at speeds of over 4 body-lengths per second, and can overcome hip-height obstacles without significantly slowing down or altering course. Shown in Figure 1-2, the design of the Sprawl family of hexapods features legs with two degrees-of-freedom each. Only the thrust direction is actuated, using pneumatic pistons embedded in the leg. The legs are attached to the body through soft flexures of visco-elastic material that allow passive rotation of the hip during running. A servomotor at each hip is used only to establish the equilibrium position of the hip joint. Pneumatic solenoid valves



**Figure 1-3.** Robot ground speed versus terrain slope for two different stride frequencies of the open-loop motor pattern. As shown, stable running occurs over a range of frequencies, although the stride frequency for maximum speed depends on the slope, which illustrates the need to understand how performance and stability change as a function of the motor pattern.

regulate air-flow into the leg pistons from a high-pressure source. The basic design uses two valves, one for each tripod of legs, mounted on the body of the robot. A tripod of legs consists of the front and hind legs from one side of the robot and the middle leg from the opposite side. The on/off valves are alternately activated according to an open-loop binary motor pattern, such that the robots achieve a trotting gait in which only one tripod is on the ground at a given time.

The body and legs of the robots are built up in layers using a rapid-prototyping process called Shape Deposition Manufacturing (Merz, et al., 1994), which allows discrete components to be embedded and soft flexures to be integrated into the body of the robot, thereby increasing robustness. More detail on the design and manufacturing can be found in Cham et al. (2001, in press) and in Bailey et al. (1999). Depending on configuration, the robots weigh between 0.25-0.33 Kg and have a length of 10-15cm. Maximum speeds range from 0.5-0.8m/sec with preferred stride frequencies of 7-10 Hz. Despite the simplicity in design and in control, stable running can be achieved over a range of stride frequencies and hip joint angles.

The Sprawl family of robots, and concurrently developed robots such as RHex (Saranli, et al., 2001), demonstrate that fast, robust locomotion can be achieved with a simple mechanical system and an open-loop motor pattern. The question now arises of how to "tune" or adjust the open-loop parameters to achieve optimal performance. While differ-

ent sets of open-loop parameters can result in stable running, the resulting behaviors may vary in performance in terms of forward velocity, or ability to reject disturbances. For example, Figure 1-3 shows a plot of the forward velocity of one of the Sprawl robots as a function of ground slope for two different stride frequencies. On level ground, the fastest locomotion is obtained with a frequency of approximately 11 Hz. But on a 20 degree slope, locomotion is considerably faster at 5 Hz than at 11 Hz. The optimal frequency for ground speed also varies somewhat from one robot to the next due to manufacturing tolerances and variations in the materials properties of the legs.

Numerical optimization techniques, such as the simplex methods utilized in Mombaur (2001) for a simulated hopper and in Weingarten et al. (2002) for RHex, have proven effective at searching the parameter space for optimal performance, resulting in behavior that may not have been achievable with manual tuning. However, such search methods must be performed offline, as they may involve implementing and evaluating combinations of the parameters that result in unstable behavior. During running in critical tasks, such behavior would be unacceptable. Furthermore, relying solely on such techniques does little to increase our basic understanding of the underlying dynamics of this class of systems.

Consequently, there is a motivation to derive insights into the basic relationships in open-loop running among performance, stability and the choice of motor pattern that not only increase our understanding of such systems, but that can also inform both the design of future robots and the development of strategies for the online adaptation of the open-loop parameters. In addition, such *a priori* knowledge of the dynamics can help reduce the sensor requirements needed for adaptation. By relating desired performance metrics such as speed and efficiency to more easily observable variables or events such as, for example, ground contact timing, such insights into the dynamics can help adaptation strategies avoid the use of complex sensors and increase their reliability. Deriving these insights and characterizing these basic relationships is the focus of this thesis.

## 1.2 Thesis Overview

Attempting to derive insights into the basic dynamics of systems like the Sprawl robots from direct observation is difficult. First, the complexity of even a simply designed robot

such as *Sprawlita*, in terms of the number of states needed to describe its motion, may confound the discovery of fundamental relationships. Second, designing an experimental setup to measure all of these states presents a significant challenge. Quantifying stability, in terms of recovery from perturbations to a steady-state trajectory, is particularly difficult, as noise and imperfections in the terrain add to the problem of introducing known disturbances and measuring how the system responds.

The approach taken in this thesis studies open-loop running through the analysis of simplified models of hopping (repeated jumping in place) and running. These models are intended to capture the basic dynamics of the interaction between passive mechanical properties and an open-loop motor pattern in a system that comes into intermittent contact with the ground. In this case, the trotting gait of the *Sprawl* robots, in which the tripods of legs come into contact with the ground one at a time, is abstracted first as a vertical one-legged hopper, and then as a planar one-legged runner. This approach is supported by the discovery that animals of different morphology and number of legs run in a similar manner that can be well described by simple mass-spring models (Full and Koditschek, 2000).

Previous studies of simplified open-loop vertical hopping models such as Ringrose (1998), Berkemeier and Desai (1998) and Komsuoglu and Koditschek (2000) have ignored either damping or gravity in their analysis, and have not fully explored the factors that affect performance and stability as the open-loop motor pattern is changed. Although the work of Mombaur (2000) explored the possibility of open-loop running in a simplified monopodal planar model and presented a framework for finding stable running trajectories, exploration of performance and stability was not given. The simplified models in this thesis include significant damping and gravity and are analyzed explicitly in terms of the effect of the open-loop motor pattern on performance and stability.

Chapter 1 has presented a context for open-loop running and given motivation for understanding its general properties. Chapter 2 presents the approach taken in the analysis of the simplified models, in which steady-state trajectories (trajectories in the state-space that repeat themselves) are first searched, and then analyzed in terms of their performance and stability. The chapter also reviews the mathematical tools used in the analysis, Poincare Maps and their Jacobians, and their special properties for the class of systems studied here.

Chapter 3 presents an analytical and experimental analysis of a vertical hopper in which actuation is controlled open loop by a motor pattern. Characterizations of performance and stability obtained through simulation and analytical expressions of the equations of motion are given and verified with experimental data of a physical vertical hopper. Chapter 4 extends these characterizations to motion in the vertical, or sagittal, plane by analyzing both a traditional general model for running, the Spring-loaded Inverted Pendulum (SLIP), and a modified model that more closely approximates running in the Sprawl robots. Chapter 5 presents experimental data of Sprawlita during running for different stride periods and system configurations that correlates with the observations made in previous chapters. These observations are then applied in a prototype adaptation algorithm that only requires binary contact information from one of the robot's feet. Experimental results of the adaptation strategy are then presented. Finally, Chapter 6 summarizes the main conclusions of the thesis, and presents an outlook for future work.

### **1.3 Contributions**

The main contributions of this thesis are:

- Novel characterizations are given of the factors that determine performance in terms of hopping height for vertical hopping and stride length and forward velocity for planar running. Performance is found to be well characterized by the net amount of work performed by the actuator on the system during a stride. Stride periods of the open-loop motor pattern that result in steady-state trajectories in which maximum work is performed correspond to stride periods that maximize hopping height and stride length. The net amount of work, in turn, is characterized by the timing of actuation activation and deactivation relative to the motion of the system.
- Exploration of the timing of actuation activation leads to an evaluation of steady-state trajectories not previously considered by analyses of hopping and running system. Traditional analyses commonly assume either conservative behavior (without energy loss), or actuation activation according to the control law implemented by Raibert (1986) in his hoppers, which initiates thrust when the leg is sensed to be at maximum compression. A significant finding is that activating thrust at maximum

compression is not always optimal in terms of performance. Exploring trajectories in which thrust is activated at other points in the cycle reveals trajectories with higher hopping heights, in the case of vertical hopping, and higher forward velocity in the case of planar running.

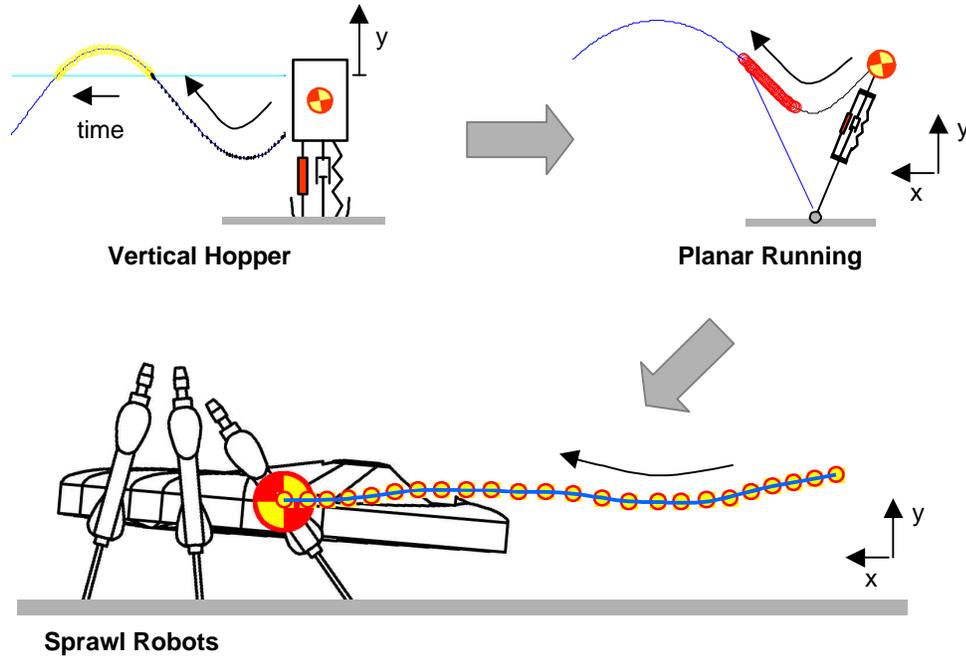
- Characterizations are given of the basic mechanisms that result in unstable steady-state trajectories under open-loop control. Unstable trajectories result when the system fails to re-synchronize its motion to the timing of the open-loop motor pattern after a perturbation. Analytical expressions and the previously established correlation between the amount of work performed on the system by the actuator and the timing of thrust activation and deactivation are used to explain this de-synchronization.
- Examination of the trade-offs in stability and performance for open-loop hopping and running systems show that optimal steady-state trajectories lie at the edge of stable behavior. The main consequence of this observation is that open-loop hopping and running systems may have to be operated at sub-optimal settings in order to avoid undesirable behavior.
- Insights into the underlying dynamics are shown to be capable of informing strategies for the adaptation, or "tuning," of the open-loop parameters. Such insights can also minimize the sensory information needed by adaptation strategies by relating performance to more easily measurable variables or events. A novel adaptation scheme is demonstrated in the hexapedal robots that adjusts the period of the open-loop motor pattern in response to changing conditions. The adaptation strategy takes advantage of the established relationships between the stride period, ground contact and performance gained throughout this work, and requires only binary contact information from one of the robot's feet.

# 2 Approach

This chapter presents the approach used to characterize the response of open-loop running systems. In the approach, simplified models are used to derive basic insights into the relationships among performance, stability and open-loop control in running. As stated previously, direct analysis of experimental platforms such as the Sprawl robots is both difficult and unlikely to yield first principles. The simplified models are examined analytically, where possible, and through simulation. Figure 2-1 shows the progression of analysis, starting from a simple vertical hopper model, followed by a more complex planar model, and finally examination of experimental data of the Sprawl robots during running. The simplified models are intended to capture the basic interaction between passive properties and an open-loop motor pattern in a system that comes into intermittent contact with the ground. The one leg of the models represents the combined effect of a tripod of legs in the robot. Since the models are planar, the one leg represents both the left and right tripods of the robot, which alternately come into contact with the ground.

Characterizing dynamic behavior through simplified models is an approach well suited for even a complex phenomenon such as locomotion. Studies of the biomechanics of running animals have shown that runners of different sizes, morphologies and number of legs exhibit similar ground reaction forces during locomotion (Full and Koditschek, 2000). Pairs and tripods of legs appear to act in concert as one "virtual leg," such that different running systems can be characterized by a simple dynamic monopodal model.

In this thesis, the approach taken for the analysis of the vertical and planar simplified models first searches for steady-state trajectories, which are trajectories in the state-space of the system that repeat themselves over time. These trajectories are found through the definition of a Poincare Map, explained later in this chapter, an approach that has the advantage that all possible trajectories can be uncovered. Finding trajectories through direct simulation (i.e. simulating the system for long periods of time and waiting for con-



**Figure 2-1.** The dynamics of open-loop running are studied through analysis of simplified vertical and planar models and experimental data of the Sprawl Robots.

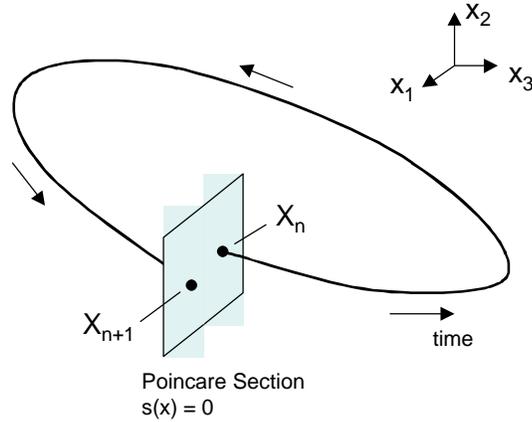
vergence to a steady-state motion) precludes finding unstable trajectories. Once found, steady-state trajectories are analyzed in terms of performance and local stability.

This approach is common in studies of locomotion systems (Kuo, 1998; Koditschek and Buehler, 1991; Nanua and Waldron, 1994; Vakakis et al., 1991). A detailed treatment of Poincare Maps and their applicability to hybrid, or variable-structure, systems such as legged running systems is given by Mombaur (2001). In this chapter, a review is given of these mathematical tools that borrows from the sources listed, with special attention given to their properties relevant to locomotion systems controlled by an open-loop motor pattern.

## 2.1 Analysis of Locomotion via Poincare Maps

Locomotion in steady-state is a cyclic, or rhythmic, phenomenon. As such, the study of locomotion has, over the past ten years, utilized tools traditionally used in the study of dynamical systems. We describe the system as a set of differential equations of the form:

$$\dot{x} = f(x, t), \quad x \in R^r \quad (2.1)$$



**Figure 2-2.** Graphical illustration of a Poincaré Map and Poincaré Section.

where  $x$  is the vector of  $r$  state variables of the system. We then consider the existence and local stability of steady-state trajectories of the system, also called steady-state orbits, that repeat themselves over time. Such an orbit  $x_0$  is defined by:

$$x_0(t) = x_0(t + T), \text{ with some } T > 0, \text{ for all } t \quad (2.2)$$

The minimum value of  $T$  is called the period of the steady-state orbit. A formal study of such cyclic trajectories first requires the definition of a function called a Poincaré Map, also called a Return Map.

### 2.1.1 Poincaré Maps and Poincaré Sections

In order to find these cyclic trajectories, we define a *Poincaré Map* as a function that transitions the state from one point in the state space to another, along a trajectory determined by the equations of motion of the system (Sastry, 1999). A Poincaré Map is associated with a *Poincaré Section*, an  $(r-1)$ -dimensional manifold in the  $r$ -dimensional state space that is assumed to be orthogonal to the studied trajectory at the point of intersection. This orthogonality requirement ensures that the trajectory intersects the Poincaré Section in a direction normal to the surface defined by the Section, such that small changes in the choice of Poincaré Map or in the trajectory do not significantly affect the results derived from it. This requirement is treated in more detailed later in the analysis. Such a Poincaré Section can be defined as a constraint equation in the state space variables:

$$s(x) = 0 \quad (2.3)$$

The Poincare Map is then defined as the mapping between two subsequent intersections of a state-space trajectory with the associated Poincare Section:

$$M: x_n(t_n) \rightarrow x_{n+1}(t_{n+1}), \text{ where } s(x_n(t_n)) = s(x_{n+1}(t_{n+1})) = 0 \quad (2.4)$$

In other words, the Poincare Map is a function  $M$  that evolves the state according to the equations of motions starting from a point in the Poincare Section (that is, a point that satisfies Equation 2.3), and returns the state the next time the trajectory intersects, or satisfies the constraint of, the Poincare Section. This concept is illustrated in Figure 2-2. Note that the Poincare Map does not necessarily map the state from one time  $t$  to the state at a time  $t + T$ , that is, subsequent intersections do not necessarily occur  $T$  time units apart.

Because the Poincare Map looks only at the state at particular instances of the trajectory, it is said to convert the study of continuous-time trajectories to a discrete-time analysis. Furthermore, since the state variables at the instant studied by the Poincare Map are constrained by the Poincare Section equation, the dimension of the discrete system is reduced to  $(r-1)$ .

Finally, for the case of autonomous systems (systems in which the equations of motion do not depend on time), it is common to define the Poincare Section based on a meaningful physical event in the state variables. For example, analyses of vertical hoppers that are modeled after Raibert's hoppers (e.g. Koditschek and Buehler, 1991, Vakakis et al., 1991) have used Poincare Sections that correspond to the point in the hopping cycle where the hopper is on the ground and has reached the bottom of its trajectory. In this example, the Poincare Section constraint equation  $s$  is defined as  $s(x) = v = 0$ , where  $v$  is the vertical velocity of the hopper's center of mass. As discussed in the next section, for the case of non-autonomous systems, such as the open-loop systems that are the focus of this thesis, special care must be taken in the definition of the Poincare Section and Map.

### **2.1.2 Poincare Maps for Clock-driven Systems**

The "clock-driven" hopping and running systems studied here are non-autonomous (that is, their equations of motion are time-dependent), as the actuators are activated according to a preprogrammed, repeating motor pattern. For non-autonomous systems, the set of "physical" state variables that one would normally choose for an autonomous system (e.g. config-

uration and velocities) do not fully describe the state of the system. As a result, time is commonly added to the state-space in some manner, either explicitly or through a "proxy" variable with appropriate time-dependent dynamics. For example, one can create a linearly-increasing state variable  $x_{r+l}$  defined as:

$$\dot{x}_{r+1} = 1, x_{r+1} \in R^1 \quad (2.5)$$

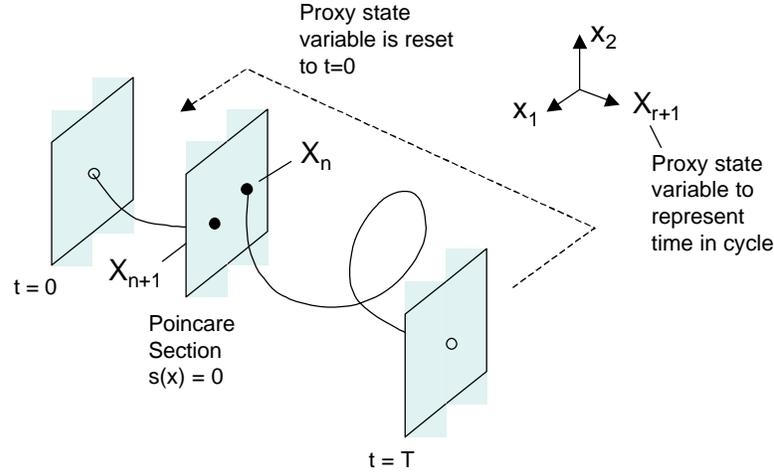
This proxy variable is then added to the set of physical state-space variables one would normally define for autonomous systems. In the systems studied here, the motor pattern is clock-driven, that is, it repeats itself over time with, for instance, period  $T$ . In this case, an alternative is to create a similar state variable that acts as a "clock" using the *mod* function:

$$x_{r+1} = t \bmod T \quad (2.6)$$

Where the *mod* function is defined such that the following conditions are satisfied (Leijen, 2001):

$$\begin{aligned} t &= T \cdot q + x_{r+1} \\ q &\in Z \text{ (an integer)} \\ |x_{r+1}| &< |T| \end{aligned} \quad (2.7)$$

This state variable then represents the time within the cycle of the repeating motor pattern, and is illustrated in Figure 2-3. The inclusion of time into the state space now lets us define Poincare Maps and Poincare Sections based on a condition in time. For example, a Poincare Map can be created that looks at the state-space variables every period  $T$ , given knowledge that the motor pattern is of period  $T$ . This case is analogous to the discretization of a continuous system with a fixed time step  $T$ , and the Poincare Map is equivalent to the discrete state transition function. In this case, the Poincare section is defined as a constraint on time explicitly or on the proxy variable, which may be advantageous in some cases. This Poincare Section can be defined using a form of the *mod* function based on the period  $T$  for the case of a proxy variable of the form of Equation 2.5, or simply as a condition such as  $x_{r+l} = 0$  (the beginning of the motor pattern cycle) for the case of Equation 2.6. In the anal-



**Figure 2-3.** Graphical illustration of the addition of a proxy "clock" state variable for non-autonomous systems driven by a repeating motor pattern.

ysis presented in this thesis, time is included in the state space as an additional state variable of the form of Equation 2.6.

### 2.1.3 Steady-state Trajectories and Fixed Points

Steady-state trajectories can now be found by solving for the *fixed points*, also called equilibrium points, of the Poincaré Map. Fixed points are the states that satisfy:

$$M(x^*) = x^* \quad (2.8)$$

Where  $M$  is the Poincaré Map function and  $x$  is the augmented vector of  $r + 1$  state variables. Thus, the solution point  $x^*$  not only represents the physical state of the system, but also the timing, or phasing, relative to the motor pattern. The function  $M$  is found explicitly in the case where the equations of motion can be integrated analytically (as in the case of the vertical hopper studied in this thesis), or through dynamic simulation (as in the case of the planar runner studied here). Though the function  $M$  can be found, solving Equation 2.8 for fixed points is most often intractable, and performed numerically. The existence of a solution point  $x^*$ , and thus the existence of a steady-state trajectory, is not guaranteed.

For a given steady-state trajectory  $x_0$ , different choices of the Poincaré Section and Map may return different solution points  $x^*$ . However, each fixed point  $x^*$  encodes the entire steady-state trajectory, as any other point on the trajectory can be found by evolving

$x^*$  according to the equations of motion. Allowing a function in time  $x_\theta(t)$  to be represented in analysis by a single vector point  $x^*$  is an advantage of the method used here.

A particular steady-state trajectory represents a time history of the state space variables and is not bound to any particular control scheme. In other words, the same steady state trajectory could be valid for systems with different control schemes (clock-driven or feedback-driven). Note that for an autonomous system, choosing a Poincare Section that looks at the state every period  $T$  will result in an infinity of solutions, as any point along the trajectory will be a fixed point of such a Poincare Map, or will result in no solutions if the true period of the steady-state trajectory is not guessed correctly.

Finally, note that the uniqueness of the trajectory  $x_\theta$  for a particular system is not guaranteed, despite the fact that Equation 2.8 and the constraint of the Poincare Section represent  $r + 1$  equations for  $r + 1$  unknowns. In addition, concurrent to steady-state trajectories that satisfy Equation 2.8, there may exist trajectories that repeat themselves after multiple intersections with the Poincare Section. Such trajectories are called *period- $X$*  trajectories, where  $X$  indicates the multiplicity of intersections with the Section. For example, a period-2 trajectory satisfies:

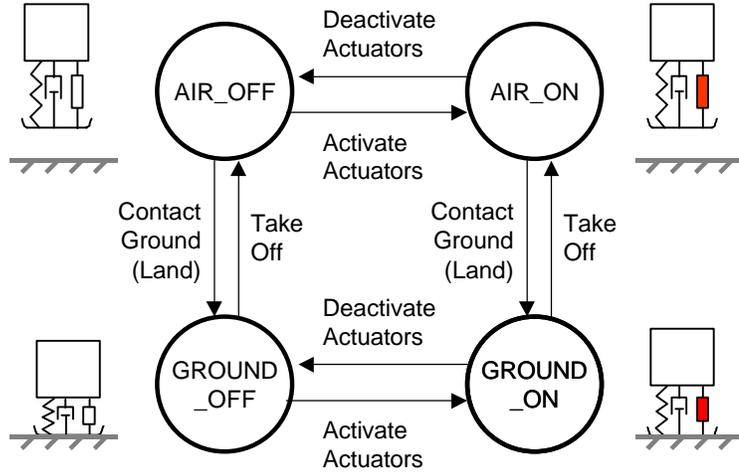
$$M(M(x^*)) = x^* \quad (2.9)$$

#### 2.1.4 Poincare Maps for Hybrid Systems

Hopping and running systems undergo distinctive changes in their dynamics as legs come into and out of contact with the ground and as actuators are activated and deactivated. As a result, locomotion systems are considered to be of "variable structure," as the definition of the function  $f$  for the equations of motion in Equation 2.1 can change due to these events. The approach taken borrows from the analysis of Hybrid, or Discrete-Continuous, systems (DeCarlo, et al., 2000) to define the dynamics and to formulate the Poincare Map.

In a Hybrid Systems framework, the dynamics are defined as a set of discrete *modes* (also called states or phases) and a set of *events* that transition the system between modes. Each mode is characterized by a set of continuous-time differential equations:

$$\dot{x} = f_i(x), \quad i = 1, 2, \dots, n_{modes} \quad (2.10)$$



**Figure 2-4.** Graphical illustration of modes and events in a general hopping system. Each node in the graph represents a mode of the system, governed by its corresponding equations of motion. The arrows represent events in the state-space or in time that cause the system to transition to another mode.

Events are triggered by conditions in the state space of the system (which may include both physical variables and time-based variables) written as constraint equations similar to the Poincare Section:

$$e_j(x) = 0, j = 1, 2 \dots n_{transitions} \quad (2.11)$$

Transitions between modes may entail instantaneous changes in the state through *reset conditions*, or there may be continuity in the state, as is assumed in the cases studied in this work. This concept is illustrated in Figure 2-4.

At this point, we restrict our study to running systems with a particular general structure of modes and events. In these systems, the changes in the dynamics of the system are due to the leg or legs touching down and lifting off the ground, and due to the actuators turning on and off. For simplicity, we assume bipedal running, such that there is only one leg on the ground at a time, or at least trotting gaits, where sets of legs that are symmetric in the sagittal plane land and take off simultaneously. Thus, we describe four general discrete modes:

GROUND\_OFF: system is in contact with the ground through a passive leg (2.12)

GROUND\_ON: system is in contact with the ground through an active leg (2.13)

AIR\_OFF: the system is airborne with the leg passive (2.14)

AIR\_ON: the system is airborne with the leg active (2.15)

and we describe the four general types of mode transitions:

AIR\_ to GROUND\_: the leg comes into contact with the ground (2.16)

GROUND\_ to AIR\_: the system loses contact with the ground (2.17)

\_OFF to \_ON: the leg actuators are activated (2.18)

\_ON to \_OFF: the leg actuators are deactivated (2.19)

The GROUND\_ON mode, or active stance mode, is characterized by work performed by the leg's actuator or actuators (the leg cannot perform work while in the air). This work could be the result of an applied control law, an open-loop forcing function that is time-dependent, or the release of stored energy (for example, through the release of a compressed spring or compressed air) and is defined as the result of a general forcing function  $u(x,t)$ . In reality, especially in animals, the transition between active and passive states may not be as abrupt as this described transition between \_ON and \_OFF modes. Nonetheless, this transition is used to generally differentiate segments during stance in which either the passive properties of the leg or work performed by actuators dominates the dynamics. To assume this transition implies that the force production in the leg is non-constant over the locomotion cycle, which is not unrealistic as a constant leg force is unlikely to yield a cyclic stride. Also, defining a transition between \_ON and \_OFF states provides an event that can be used to measure the relative phasing, or relative time difference, of the forcing function with respect to the motion of the system, which is punctuated by the GROUND\_/AIR\_ transitions.

In open-loop running systems, transitions between GROUND and AIR modes are triggered by events in the physical state space (e.g. when the foot comes into contact with the ground or when the leg has reached its maximum length) while transitions between ON and OFF modes are effected by the open-loop motor pattern, that is, by events in the proxy state variable used to represent the cycle time. As a result of the existence of multiple modes and multiple possible transitions from each mode, a particular trajectory may traverse the mode diagram of Figure 2-4 in any number of ways. Steady-state trajectories will eventually return to their starting mode and repeat their traversal through the diagram,

but the exact sequence of mode changes that occur during a cycle can take different forms. Because these multiple sequences can exist concurrently as different steady-state trajectories for a given set of system parameters, it is important to consider all of them to understand the system's possible behaviors. Practically, we must limit the number of co-existing sequences studied.

The approach taken here is to enumerate a set of possible sequences that are considered to be "normal" running. We limit the study to period-1 steady-state trajectories in which the airborne phase is followed by a passive stance mode, followed then by the activation of the leg actuator and finally by the subsequent deactivation of the actuator shortly before or shortly after take-off into the next airborne mode. The two sequences that fit this description are:

$$\{\text{AIR\_OFF, GROUND\_OFF, GROUND\_ON, AIR\_ON}\} \quad (2.20)$$

$$\{\text{AIR\_OFF, GROUND\_OFF, GROUND\_ON, GROUND\_OFF}\} \quad (2.21)$$

As will be shown in later chapters, the difference between these two sequences, that is, deactivating the leg actuator before or after take-off, can have significant consequences in terms of performance and stability.

Having enumerated the sequences, a Poincare Map is formulated for each sequence in a given system by integrating the equations of motion within each mode, and solving for the boundary conditions within each mode, that is, finding the exact time and state that the trajectory triggers the mode-changing events according to the assumed sequence of mode transitions. Note that multiple fixed point solutions corresponding to different mode sequences may exist for a given set of open-loop parameters.

## 2.2 Resonance in Running

Having defined steady-state solutions and their associated trajectories for a given sequence of mode events in locomotion, we now take an aside to discuss a subject of relevance later on in the analysis of the simplified models. Although the concept of resonance is easily understood at an intuitive level, an exact definition in the context of running systems does not come as easily. Resonance is commonly defined in linear systems in the context of a one-degree-of-freedom mass-spring-damper system and is described as the situation in

which a sinusoidal forcing function is applied at the same frequency as the system's natural frequency and for which a "small input" results in a "large output."

Defined as such, it is easy to see its desirability in running systems. Achieving a form of resonance in running suggests minimizing the input, or work, necessary to maintain locomotion. It has been observed in biomechanical studies that animals have a preferred stride frequency when running. Presumably, given no objective other than mobility, the animal chooses a stride that maximizes output for a given input level.

Unfortunately, a simple definition of the "natural frequency" as the square root of stiffness over mass is not appropriate given the non-linearities present in even simple models for locomotion. Furthermore, such a traditional definition is less meaningful for running systems that are controlled by non-periodic forcing functions such as feedback-based controllers. Finally, given the multiple degrees of freedom in running systems, defining resonance in terms of the magnitude of output of a particular degree of freedom will ultimately limit our ability to understand the entire system.

A more general definition looks at the work performed by the actuator during a cycle of locomotion. This measure of work performed is defined as the integral of the dot product of the forcing function  $u(x,t)$  with the state's generalized velocities  $v(t)$  over the duration of the stride:

$$\int_{t_0}^{t_0+T} u(x, t) \cdot v(t) dt \quad (2.22)$$

where  $T$  is the period of the particular steady-state trajectory.

A particular solution or steady-state trajectory will entail a relative phasing between the motion of the system and the open-loop motor pattern in order for the steady-state conditions to be met. Given such a phasing, the forcing function may perform both positive and negative work. For example, in the case of a simple linear mass-spring-damper system that is excited by a high-frequency forcing function, the phasing is such that the forcing function acts to decelerate the mass for a certain part of the cycle.

A situation may arise in which the phasing of the steady-state trajectory is such that the net amount of work performed by the forcing function is maximized. Such a situation

can be defined as "resonance." In this case, resonance is a relative quantity between steady-state solutions in the set that results from varying the open-loop parameters (e.g. stride period). This attention to the net amount of work performed will have significance in later chapters which address the question of whether "resonance," or the maximization of work performed in a locomotion cycle, is related to performance metrics of interest such as hopping height and forward velocity.

## 2.3 Stability of Steady-state Trajectories

When started at any point along a steady-state trajectory, the system will remain in that trajectory. However, perturbations or errant initial conditions will place the system away from such a trajectory. Whether the system returns to the nearby steady-state trajectory or moves away from it is determined by the trajectory's *local stability*, which is defined in this section. References for the following sections include (Sastry, 1999 and Mombaur, 2001).

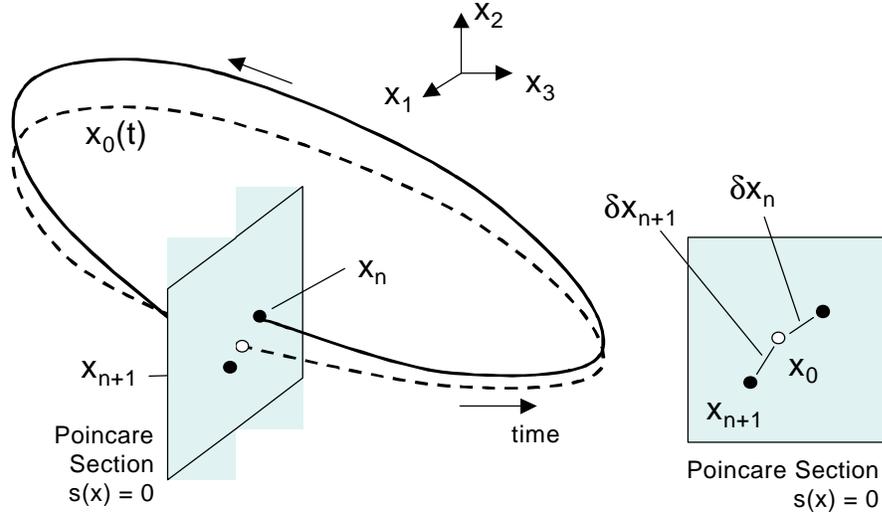
### 2.3.1 Local Stability and Floquet Multipliers

A steady-state trajectory  $x_0(t)$  of a system defined by the non-autonomous differential equations,

$$\dot{x} = f(x), x \in R^{r+1} \quad (2.23)$$

(where  $x$  is the augmented vector of state variables that include a proxy variable for time) is said to be Lyapunov-stable if, for every  $\epsilon$  there is a  $\delta$  such that if for all solutions  $x(t)$  with  $\|x(t_0) - x_0(t_0)\| < \delta$  then  $\|x(t) - x_0(t)\| < \epsilon$  for all  $t$ . Such a steady-state trajectory  $x_0(t)$  is said to be asymptotically-stable in the sense of Lyapunov if:

$$\lim_{t \rightarrow \infty} \|x(t) - x_0(t)\| = 0 \quad (2.24)$$



**Figure 2-5.** Graphical illustration of the Jacobian, which maps perturbations about a steady-state trajectory,  $x_0(t)$ , between subsequent intersections with the Poincaré Section.

Note that since the state space includes the proxy variable for time, this implies phase stability, that is, synchronization with the steady-state trajectory in terms of the open-loop motor pattern.

A common tool for determining stability of steady-state trajectories is the use of Floquet analysis (Sastry, 1999). Floquet analysis looks at the multi-variable derivative of a Poincaré Map, called the Jacobian  $J$ , evaluated at the fixed point associated with that Map:

$$J_M(x^*) = \left. \frac{\partial M}{\partial x_n} \right|_{x^*} = \left. \frac{\partial x_{n+1}}{\partial x_n} \right|_{x^*} \quad (2.25)$$

This multivariable derivative is a linear transformation that maps infinitesimal disturbances about the steady-state trajectory at one intersection with the Poincaré section to disturbances at the next intersection, as illustrated in Figure 2-5. As such, the local stability of the steady-state trajectory is given by the eigenvalues  $\lambda_i$ , also called Floquet multipliers, of this Jacobian matrix. The steady-state trajectory is said to be locally stable if all of the eigenvalues  $|\lambda_i| \leq 1$ , asymptotically stable if  $|\lambda_i| < 1$  and unstable if at least one eigenvalue  $|\lambda_i| > 1$ . The applicability of Lyapunov's first method to hybrid periodic systems such as the running systems studied here is shown in (Mombaur, 2001).

Note that due to the constraint imposed on the state at  $t_n$  and  $t_{n+1}$  by the Poincare Section, the Jacobian as defined in Equation 2.25 will have one eigenvalue equal to 0. For subsequent analysis, we will denote  $\tilde{J}_M$  as the Jacobian with the appropriate row and column removed for one of the state variables constrained by the Section. For example, if the Poincare Section were described by  $x_I = 0$ , then the first row and column would be removed

### 2.3.2 Derivation of Jacobian for Hybrid Systems

An interesting and useful characteristic of the Jacobian for a periodic system that undergoes mode changes is that it can be shown to be the product of individual "sensitivity" matrices (Mombaur, 2001), which map disturbances between the boundaries of each mode along the steady-state trajectory:

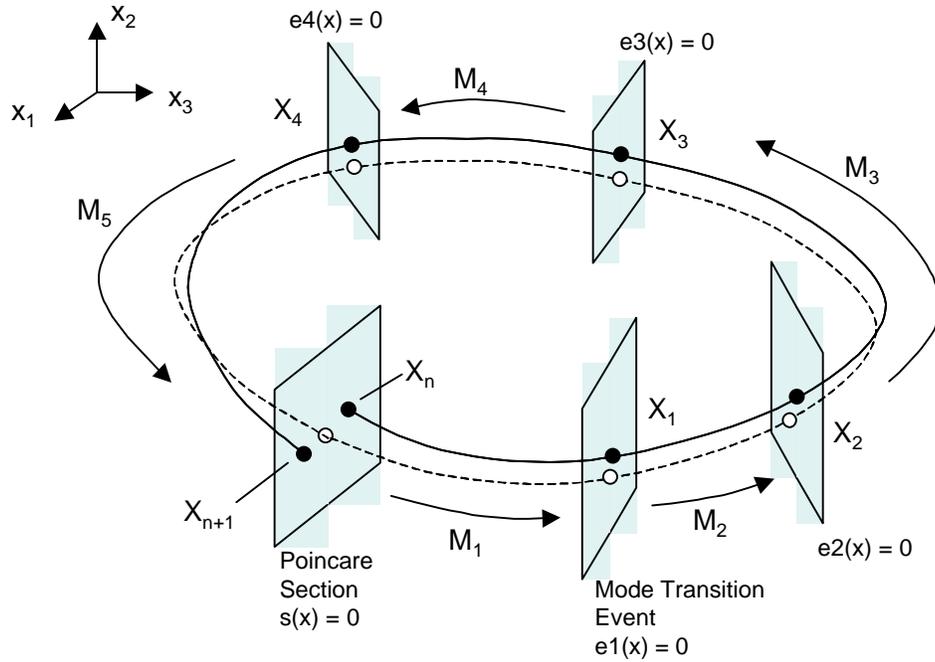
$$\tilde{J}_M(x^*) = \tilde{J}_1(x^*) \cdot \tilde{J}_2(x^*) \dots \tilde{J}_m(x^*) \quad (2.26)$$

where  $m$  is the number of modes along the steady-state trajectory. This concept is illustrated in Figure 2-6. As shown, the first section of the steady-state trajectory starts at the Poincare Section,  $s(x)=0$ , and ends at the first mode transition, given by that event's constraint equation  $e_1(x)=0$ . Subsequent sections along the trajectory are characterized by a particular mode and are bound by the event constraints equations  $e_i(x)=0$ . Like the Jacobian for the overall Poincare Map, the individual sensitivity matrices are the multivariable derivatives of the maps that transition the state within each mode, from one boundary to the next:

$$\tilde{J}_i(x^*) = \left. \frac{\partial \tilde{M}_i}{\partial \tilde{x}_{i-1}} \right|_{x^*} = \left. \frac{\partial \tilde{x}_i}{\partial \tilde{x}_{i-1}} \right|_{x^*}, \quad i = 1, 2 \dots m \quad (2.27)$$

$$\tilde{M}_i : \tilde{x}_{i-1}(t_{i-1}) \rightarrow \tilde{x}_i(t_i) \quad (2.28)$$

Here,  $\tilde{x}_i(t_i)$  is the reduced state vector at the mode boundaries, where an appropriate degree of freedom constrained by the event constraint equation  $e_i(x)=0$  has been removed. For example, if  $e_i(x) = x_I = 0$ , then the vector  $\tilde{x}_i(t_i)$  is the state vector  $x$  without the state variable  $x_I$ .

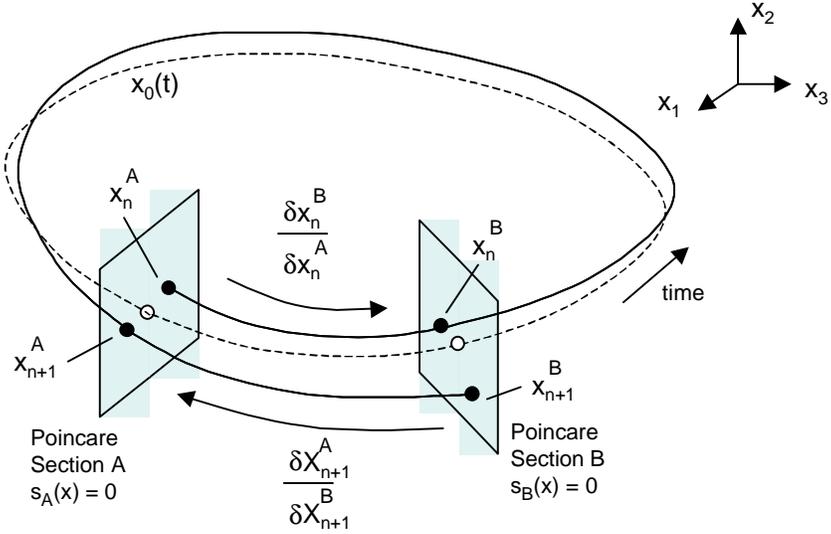


**Figure 2-6.** Graphical illustration of sensitivity matrices. Along a given steady-state trajectory (shown dotted), there may exist multiple mode transitions. The Jacobian for the Poincare Map can be written as the product of a set of "sensitivity" matrices, which are the multivariable derivatives of the maps  $M_i$  between the states at the mode transitions, given by the intersections with the corresponding mode boundaries.

This property of the Jacobian allows it to be computed in steps by first computing the individual sensitivity matrices and then using the product to find the Poincare Map's over-all Jacobian. For hybrid systems such as the ones studied here, this approach allows computed sensitivity matrices to be "reused" for Poincare Maps that have common modes in their mode sequence.

For the specific case of a general hybrid system with linear modes, a general formula for computing an arbitrary individual sensitivity matrix with either state- or time-based linear boundary constraints equations can be derived, as detailed in Appendix A. To the best of the author's knowledge, this derivation has not been presented before. This general formula is used in Chapter 3 for the vertical hopping model.

For the case of equations of motion which cannot be integrated analytically, the Jacobian is computed numerically by introducing perturbations along a steady-state trajectory and measuring the resulting deviations. This method is used in the planar hopping model analyzed in Chapter 4.



**Figure 2-7.** Graphical illustration of two Poincaré Maps for a given steady-state trajectory. The Jacobians for both Poincaré Maps are related by a similarity transformation.

### 2.3.3 Jacobian Similarity Transformation

Another interesting and useful property of the Jacobian is its invariability in terms of its eigenvalues to the choice of Poincaré Map (Mombaur, 2001). Although different Poincaré Sections and Maps will result in different Jacobian matrices, they will all share the same eigenvalues. That is, conclusions made about the local stability of a steady-state trajectory are independent of the point along the trajectory used for the Poincaré Map, and independent of whether the Poincaré Section is time- or state-based.

To show this, we define two Poincaré Maps  $\tilde{M}_A$  and  $\tilde{M}_B$  (remember the tilda notation implies that the maps include a minimum set of state variables given the constraints of the Poincaré Sections), as shown in Figure 2-7. Each Poincaré Map returns subsequent intersections with its Poincaré Section  $\tilde{M}_A : \tilde{x}_n^A \rightarrow \tilde{x}_{n+1}^A$  and  $\tilde{M}_B : \tilde{x}_n^B \rightarrow \tilde{x}_{n+1}^B$ . Similar to the sensitivity matrices in the previous sections, we can define the following two multi-variable matrices:

$$\left. \frac{\partial \tilde{x}_n^B}{\partial \tilde{x}_n^A} \right|_{x^*} \quad \text{and} \quad \left. \frac{\partial \tilde{x}_{n+1}^A}{\partial \tilde{x}_{n+1}^B} \right|_{x^*} \quad (2.29)$$

which map disturbances between the two Poincare Sections along the steady-state trajectory. Note that  $\left. \frac{\partial(\tilde{x}_{n+1}^A)}{\partial(\tilde{x}_{n+1}^B)} \right|_{x^*}$  maps disturbances back in time between Poincare Sections B and A. Thus, we can rewrite the Jacobian based on  $\tilde{M}_A$  as:

$$\tilde{M}_A = \left. \frac{\partial \tilde{x}_{n+1}^A}{\partial \tilde{x}_n^A} \right|_{x^*} = \left. \frac{\partial \tilde{x}_{n+1}^A}{\partial \tilde{x}_{n+1}^B} \right|_{x^*} \cdot \left. \frac{\partial \tilde{x}_{n+1}^B}{\partial \tilde{x}_n^B} \right|_{x^*} \cdot \left. \frac{\partial \tilde{x}_n^B}{\partial \tilde{x}_n^A} \right|_{x^*} = \left. \frac{\partial \tilde{x}_{n+1}^A}{\partial \tilde{x}_{n+1}^B} \right|_{x^*} \cdot \tilde{M}_B \cdot \left. \frac{\partial \tilde{x}_n^B}{\partial \tilde{x}_n^A} \right|_{x^*} \quad (2.30)$$

For infinitesimal disturbances about the steady-state trajectory, the two sensitivity matrices defined above are the inverse of each other:

$$\left. \frac{\partial \tilde{x}_{n+1}^A}{\partial \tilde{x}_{n+1}^B} \right|_{x^*} = \left( \left. \frac{\partial \tilde{x}_n^B}{\partial \tilde{x}_n^A} \right|_{x^*} \right)^{-1} \quad (2.31)$$

Which makes the following relationship,

$$\tilde{M}_A = \left. \frac{\partial \tilde{x}_{n+1}^A}{\partial \tilde{x}_{n+1}^B} \right|_{x^*} \cdot \tilde{M}_B \cdot \left. \frac{\partial \tilde{x}_n^B}{\partial \tilde{x}_n^A} \right|_{x^*} \quad (2.32)$$

a similarity transformation, which preserves the eigenvalues:

$$\tilde{M}_A = C^{-1} \cdot \tilde{M}_B \cdot C \quad (2.33)$$

This property of the Jacobian allows one to evaluate local stability using Poincare Maps that do not necessarily sample the trajectory at  $T$  intervals. It also allows one to compute the Jacobian based on a convenient Poincare Map, and easily transform it to a Jacobian for another Map that may provide more insight when analyzed directly.

## 2.4 Chapter Summary

The approach used in this thesis for the analysis of open-loop running systems was presented. This approach looks at simplified models which are intended to capture the basic dynamics that result from the interaction of passive properties with an open-loop motor pattern. These analyses are used to derive insights into the basic relationships that determine performance and stability, and are later verified with experimental data of physical robots. Analysis of these models is performed in terms of steady-state trajectories, or orbits. These

trajectories are found by defining a Poincare Map which maps subsequent intersections with its corresponding Poincare Section, and which reduces the analysis to a discrete system with a reduced state space. The fixed points of the Poincare Map, that is, points which map unto themselves, represent the entire steady-state trajectory. For open-loop motor patterns, the system is non-autonomous, and time must be added to the set of physical state variables either explicitly or through a proxy state variable. For hybrid, or discrete-continuous, systems, steady-state trajectories may traverse the different modes of the system in multiple sequences. The approach taken here is to enumerate sets of sequences of mode transitions considered to be of interest and to formulate the Poincare Maps based on these sequences.

The local stability of these steady-state trajectories is evaluated using Floquet Analysis via the computation of the multi-variable derivative of the Poincare Map. Special properties of this Jacobian for the systems studied here show invariability of this analysis to the choice of Poincare Map, whether it is state- or time-based. This property allows us to formulate a Poincare Map for open-loop running systems based on either the moment of thrust activation, or the moment of landing, with invariance in the conclusions made about stability. In the analytical analysis of the vertical hopper, formulating the Poincare Map in each of the two manners allows analytical results which would not have been possible with just one formulation.

These general mathematical concepts and tools are utilized in subsequent chapters to analyze the behavior of simplified models of open-loop running.

# 3 Open-loop Vertical Hopping

This chapter begins the study of the underlying dynamics in open-loop running systems by analyzing a one-degree-of-freedom vertical hopper model. A vertical hopper consists of a mass that is constrained to move only in the vertical direction and that comes into intermittent contact with the ground through to a telescoping leg that contains both passive and active force-producing elements. This simplified model is intended to shed light on basic relationships among system energy, actuator timing and stability and to provide insight into the circumstances under which a stable, steady-state hopping cycle can be achieved with an open-loop control scheme. This chapter first surveys related work in the analysis of simplified vertical hopping models and then describes the model used here and its important differences to previous work. Next, analytical expressions of the Poincare Maps and associated Jacobians are derived for hopping trajectories of the model that correspond to the two sequences of modes of interest in normal hopping. The chapter then analyzes the continuum of solutions to the steady-state condition, which leads to a characterization of the basic mechanisms that determine performance in terms of vertical hopping height, and stability. Finally, these observations are verified with experimental data of a physical, pneumatically-actuated, vertical hopper called the "Dashpod."

## 3.1 Related Work

Variations on a basic vertical hopping model have been examined by several investigators, including Raibert (1986), Koditschek and Buehler (1991), Vakakis et al, (1991), Ringrose (1997), Berkemeier and Desai (1998) and Komsuoglu and Koditschek (2000). Despite its apparent simplicity, the one-legged vertical hopper exhibits a rich set of dynamic behaviors including stable and unstable periodic motion.

Raibert's mechanical hopper (Raibert, 1986) uses a double-acting pneumatic cylinder and an actuated hip to produce forward motion. The cylinder acts as an air spring when its valves are closed, which can be modeled as a non-linear spring. Thrust is initiated in a closed-loop fashion when the leg is sensed to be at maximum compression, at which point the valves are opened and the pneumatic cylinder produces a force equal to the input pressure times the area of the piston. Vertical and horizontal motions are assumed to be decoupled and vertical hopping height was found to be a function of the thrust duration. Simplified one-DOF hopper models based on this closed-loop thrust activation and non-linear spring were analyzed by Koditschek and Buehler (1991), who show that this closed-loop strategy is stable, and by Vakakis et al (1991), who show that chaotic, or non-period-1 behavior can occur for certain sets of system parameters.

Ringrose (1997) showed that a vertical hopper can maintain stable hopping without sensory feedback, using a linear actuator in series with a spring, and a damper in parallel with the actuator/spring combination. For analysis, Ringrose uses a simplified model in which thrust is applied through an impulsive change in leg length at fixed intervals in time such that thrust initiation occurs before maximum compression of the spring. Ringrose analyzes stability of this vertical hopper by decoupling hopping height stability and phase-stability (motion of the hopper relative to the cycle of the open-loop motor pattern), in both cases applying simplifications that assume low damping. In his analysis, it was found that the hopper is unstable if thrust initiation occurs at maximum compression, and it is assumed that trajectories in which thrust starts after maximum compression do not exist. This instability is attributed to the following effect: if thrust initiation occurs at maximum compression, then errors in the time that thrust is initiated will result in corresponding errors in the hopper's airborne phase that lead to even larger errors the next time that thrust is initiated.

Berkemeier and Desai (1998) compare Raibert's method of applying thrust at maximum compression, Ringrose's open loop control, and a proposed "adaptive periodic forcing" method that adjusts the period of the open loop control based on the velocity at the time that thrust is applied. In their analysis, they use a hopper with a spring and damper in parallel and an actuator that changes the neutral point of the spring. Their analysis shows that their hopper reaches maximum hopping height when the force is applied at maximum

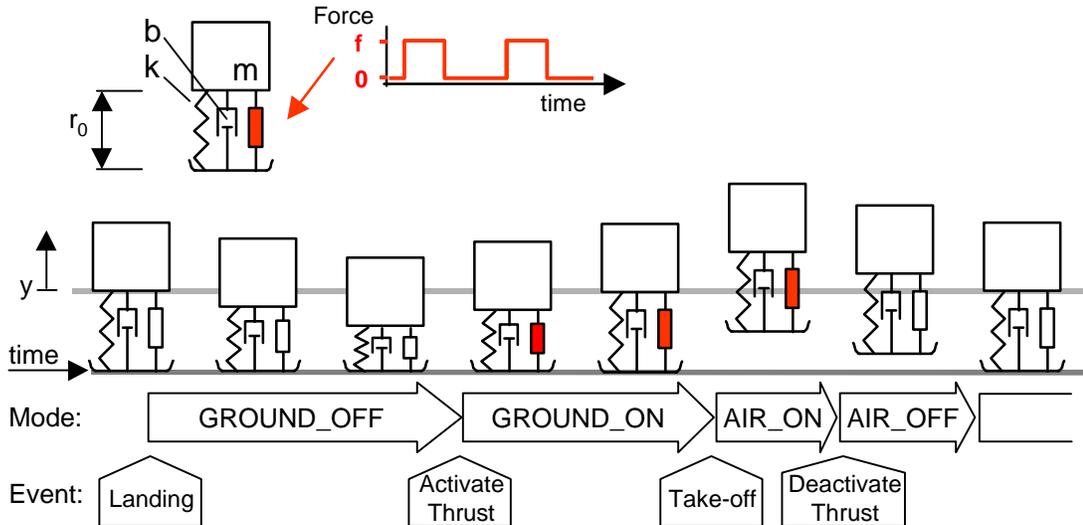
spring compression. Using perturbation methods that assume low values of damping, they also show that the motion is stable, without feedback, when the force is applied prior to the maximum compression of the spring and unstable when the force is applied after maximum compression.

Komsuoglu and Koditschek (2000) analyze the stability of a similar open-loop one-DOF hopper in which thrust is effected by clock-driven changes in the leg spring's stiffness. Neglecting gravity for simplification, they find conditions for stability, which include the necessary presence of viscous friction.

Despite this extensive history of analysis, several questions remain in terms of the basic dynamics of an open-loop vertical hopper: How does performance vary as a function of the open-loop motor pattern? What are the trade-offs in stability? What is the effect of including both damping and gravity in the analysis? What are insights that can lead to strategies for tuning of the motor pattern for optimal hopping using simple sensors?

## **3.2 Vertical Hopper Model**

The physical implementation of the Sprawl robots requires that the minimal damping assumptions made in previous models be reconsidered. In insects, and in the Sprawl family of robots, viscoelastic materials dissipate substantial amounts of energy per cycle. Typical dimensionless damping ratios for these systems are on the order of 0.3 (Garcia et al., 2000). The model studied here includes significant damping (damping ratio of 0.2), and gravity (of similar magnitude as the thrust force). Both of these assumptions make finding closed forms for the steady-state solutions intractable, though it is still possible to find analytical expressions for the multi-variable Jacobian. In addition, this analysis considers the case where thrust is deactivated before take-off, which was not considered by either Ringrose (1998) or Berkemeier and Desai (1998). In this section, the details of the model are first described, followed by the derivation of the Poincare Maps and associated Jacobians. Finally, results of exploring the continuum of solutions for the steady-state condition are presented. Subsequent sections characterize performance and stability based on these results.



**Figure 3-1.** The vertical hopping model used for analysis. The hopper’s leg consists of a spring, a damper and a force element which is active according to a binary motor pattern. The figure shows a sample trajectory of the hopper, the different modes that it goes through, and the events that trigger the transitions between the modes.

### 3.2.1 Model Description

The simplified vertical hopping model used in this analysis is illustrated in Figure 3-1. As shown, the model consists of a point mass,  $m$ , attached to a massless leg which contains in parallel a damping element,  $b$ , a linear spring  $k$ , and a force source,  $f(t)$ . The leg has a nominal rest length of  $r_0$ . The leg comes into contact with the ground (landing) when  $y=0$  (the ground is placed at  $y=-r_0$ ). Take-off is assumed to occur when the leg length exceeds the nominal rest length. With this assumption, the presence of damping in the leg model may cause the ground reaction force to be negative for a short period of time. However, this assumption was found to cause only small differences in the steady-state solutions for the range of damping and mass values used here.

In simulations of the model, the force source is activated at fixed intervals in time with period  $\tau$  and thrust duration  $t_{on}$ , according to the open-loop motor pattern. For description of the model and subsequent analysis, we introduce the time duration variable  $t_{off}$ , which is the length of time between initial landing of the leg and the instant that thrust is activated. This time duration can be used as a measure of the phasing of the motion of the system, given by touchdown, with respect to the open-loop motor pattern.

The modes and mode transition events of this hybrid, or variable structure, system were generally described in the previous chapter by Equations 2.12 to 2.19. The specific differential equations for each of the modes can be written, in normalized coordinates, as:

$$\dot{X} = AX + B \quad (3.1)$$

where A and B are defined as:

$$A = \begin{bmatrix} 0 & 1 \\ -\omega^2 & -2\zeta\omega \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ f(t) - 1 \end{bmatrix} \quad \left. \vphantom{\begin{matrix} A \\ B \end{matrix}} \right\} \quad (y \leq 0) \quad (3.2)$$

during the stance, or ground-contact, mode, and:

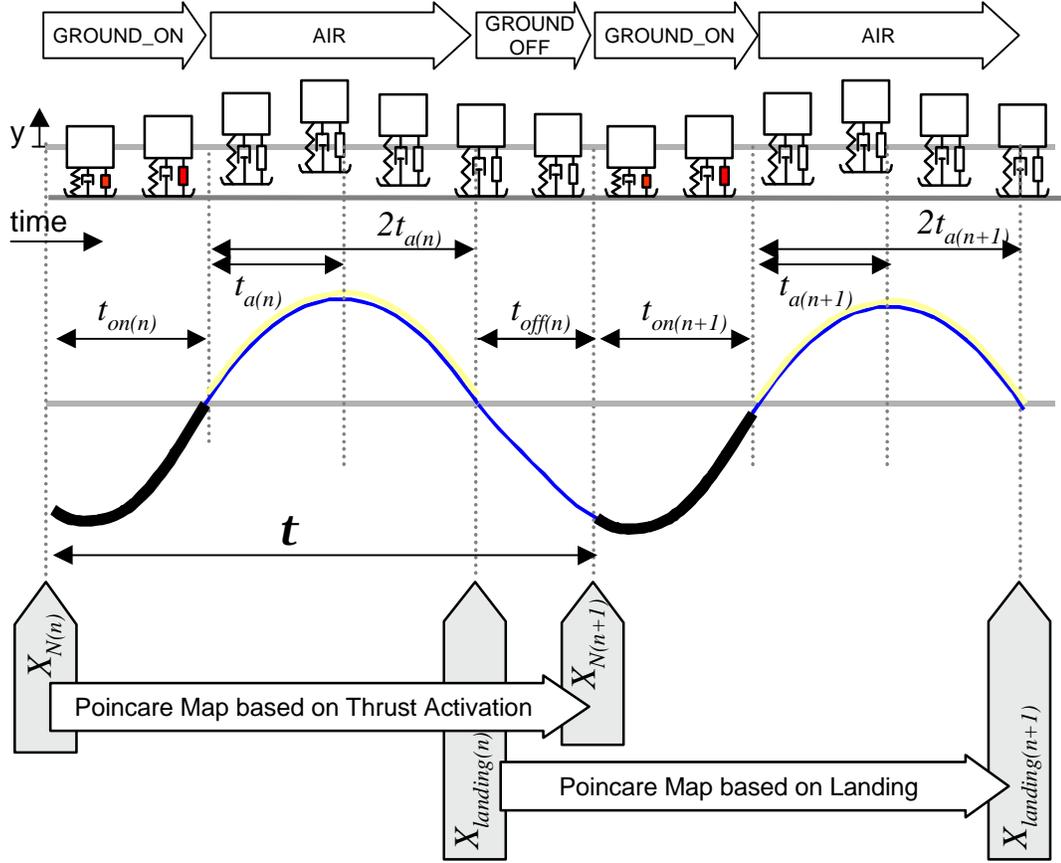
$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \quad B = \begin{bmatrix} 0 \\ -1 \end{bmatrix} \quad \left. \vphantom{\begin{matrix} A \\ B \end{matrix}} \right\} \quad (y > 0) \quad (3.3)$$

during the airborne, or ballistic, mode.  $\omega$  and  $\zeta$  are the natural frequency and damping ratio of the mass-spring-damper system. The thrust force  $f(t)$  is determined by the open-loop motor control pattern:

$$f(t) = \begin{cases} f & t_{off} < t \leq t_{off} + t_{on} \\ 0 & otherwise \end{cases} \quad (3.4)$$

where  $f$  is the normalized thrust magnitude (force normalized by weight). Here, the time  $t$  is reset to  $t=0$  when  $t$  reaches  $\tau$ . Since each of the modes is governed by linear differential equations, we can integrate to find the time solutions of the state vector  $X(t)$  for the four modes:

$$\begin{aligned} \text{AIR\_ON/_OFF: } X(t) &= \begin{bmatrix} 1 & t \\ 0 & 1 \end{bmatrix} X_0 - \begin{bmatrix} t^2/2 \\ t \end{bmatrix} \\ \text{GROUND\_ON: } X(t) &= e^{At} (X_0 - X_{eon}) + X_{eon} \\ \text{GROUND\_OFF: } X(t) &= e^{At} (X_0 - X_{eoff}) + X_{eoff} \end{aligned} \quad (3.5)$$



**Figure 3-2.** Illustration of a sample time history of the vertical hopper. The figure shows the two possibilities for formulating the Poincare Map used in analysis: a Map based on the state at thrust activation, and a Map based on the velocity and time at landing.

Here,  $X_o$  is the state at the beginning of each mode and  $X_{eon}$  and  $X_{eoff}$  are the equilibrium states for each of the stance modes:

$$X_{eon} = \begin{bmatrix} \frac{f-1}{w^2} \\ 0 \end{bmatrix} \quad X_{eoff} = \begin{bmatrix} \frac{1}{w^2} \\ 0 \end{bmatrix} \quad (3.6)$$

### 3.2.2 Derivation of Poincare Map for Steady-state Solutions

As stated in the previous chapter, the approach to finding the Poincare Map for a hybrid system such as this one involves enumerating the possible sequences of modes in periodic orbits of interest. In this case, we consider the two hopping behaviors characterized as "Long Thrust" and "Short Thrust." In "Long Thrust," we assume that the hopper first lands,

then activates thrust during stance, and that the thrust application duration is long enough to continue until or past take-off, such that the mode sequence is as given by Equation 2.20, here repeated for completeness:

$$\text{"Long Thrust"} : \{\text{AIR\_OFF}, \text{GROUND\_OFF}, \text{GROUND\_ON}, \text{AIR\_ON}\} \quad (3.7)$$

To derive the Poincare Map, we take advantage of the facts that the take-off velocity is the negative of the landing velocity and that this velocity is, in normalized coordinates, equal to  $t_a$ , or one half the total airborne time. We also take advantage of the fact that the total duration of the modes must equal  $\tau$ , the period of the open-loop motor pattern. One Poincare Map that can be easily found is formulated about the instant that thrust is activated. As illustrated in Figure 3-2, this allows the variables mapped to be the physical state variables, which is convenient since the time solutions are linear functions of these state variables. Thus, nesting the time solutions for the individual modes in the "Long Thrust" sequence, we obtain:

$$X_{N(n+1)} = M(X_{N(n)}) = X_{eoff} - e^{A(\tau - 2t_{a(n)} - t_{on(n)})} (X_{eon} + X_{eoff}) \dots \quad (3.8)$$

$$- e^{A(\tau - 2t_{a(n)})} (X_{N(n)} - X_{eon})$$

Note that the  $(n)$  notation indicates that the variable corresponds to the  $n^{\text{th}}$  trajectory, as illustrated in Figure 3-2. Here,  $t_{on}$  is a variable, which is constrained by the take-off event constraint equation:

$$X_{takeoff(n)} = e^{At_{on(n)}} (X_{N(n)} - X_{eon}) = \begin{bmatrix} 0 \\ t_{a(n)} \end{bmatrix} \quad (3.9)$$

Solving this system involves four unknowns ( $X_N$ ,  $t_{on}$  and  $t_a$ ) in four equations (Equations 3.8 and 3.9). Alternatively, it becomes more convenient to define a Map about the instant that landing, or touchdown, occurs, as shown in Figure 3-2. Here, the variables mapped are the landing velocity,  $t_a$ , and the time duration  $t_{off}$ . Using the fact that  $X_{takeoff} = -X_{landing} = [0 \ t_a]^T$ , the Map becomes:

$$X_{landing(n+1)} = -e^{A(t_{on(n+1)} + t_{off(n)})} (X_{landing(n)} - X_{eoff}) - \dots \quad (3.10)$$

$$e^{At_{on(n+1)}} (X_{eoff} - X_{eon}) - X_{eon}$$

which can be rewritten as:

$$\begin{aligned}
& X_{takeoff} + e^{A(\tau - 2t_{a(n+1)})}(X_{takeoff} + X_{eoff}) + \dots \\
& e^{A(\tau - 2t_{a(n+1)} - t_{off(n)})}(X_{eon} - X_{eoff}) - X_{eon} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}
\end{aligned} \tag{3.11}$$

Thus, to find fixed point solutions, we can set the period  $\tau$ , and numerically solve the above two equations for the two unknowns  $[t_{off} \ t_a]^T$ . Note that given the steady-state condition, the subscripts  $(n)$  and  $(n+1)$  can be ignored. Note also that this is not the complete Map based on touchdown, as an expression for  $t_{off(n+1)}$  is missing. However, such an expression is not needed to find the fixed point, as we already took advantage of the fact that the total duration of the modes must equal  $\tau$  in steady-state.

The second case studied is the sequence of mode changes in which the hopper lands and activates thrust during stance, but ends thrust application before take-off, such that the mode sequence is:

$$\begin{aligned}
& \text{"Short Thrust"} : \{AIR\_OFF, GROUND\_OFF, GROUND\_ON, \\
& GROUND\_OFF\}
\end{aligned} \tag{3.12}$$

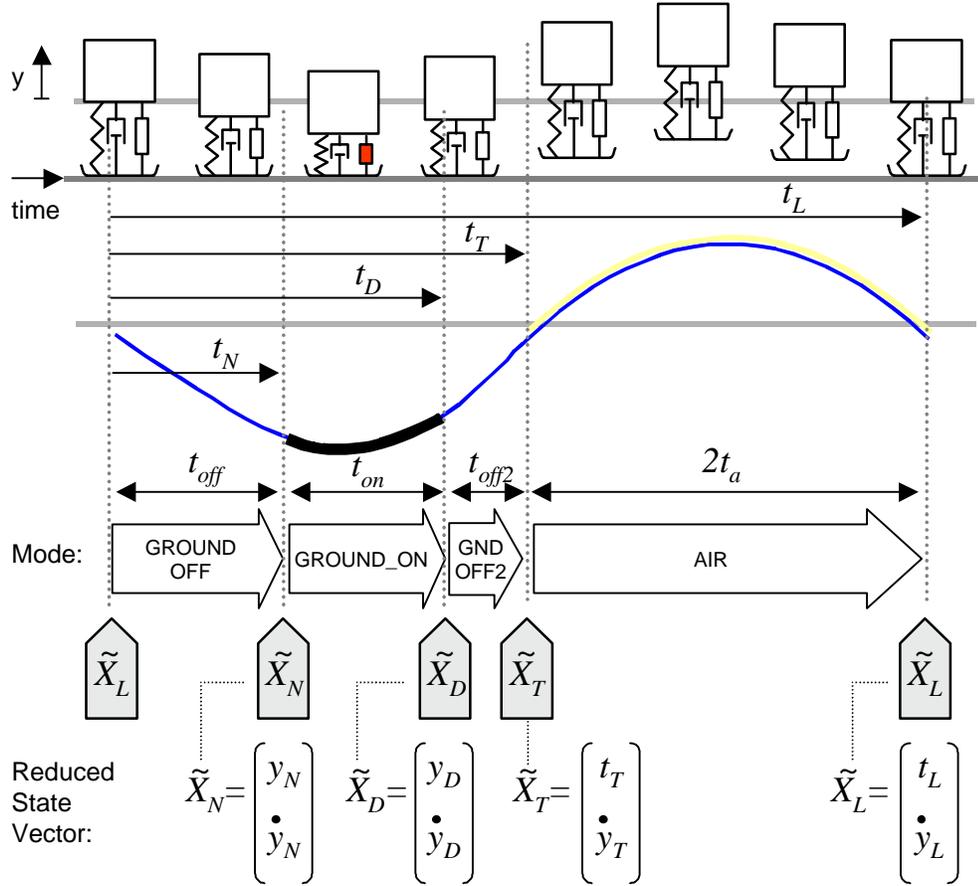
In this case, the variable  $t_{on}$  is a known constant, and the Poincare Map based on the instant of thrust application is again found by nesting the time solutions, taking advantage of the constraint that the total time duration must equal to  $\tau$ :

$$\begin{aligned}
X_{N(n+1)} = M(X_{N(n)}) = X_{eoff} - 2e^{A(\tau - 2t_{a(n)} - t_{on} - t_{off(n)})}X_{eoff} - \dots \\
e^{A(\tau - 2t_{a(n)} - t_{on})}(X_{eon} - X_{eoff}) - e^{A(\tau - 2t_{a(n)})}(X_{N(n)} - X_{eon})
\end{aligned} \tag{3.13}$$

Similar to the "Long Thrust" case, finding the fixed point becomes easier if the Poincare Map is formulated about the instant that touchdown occurs, resulting in:

$$\begin{aligned}
-X_{landing(n+1)} = e^{A(\tau - 2t_{a(n+1)})}(X_{landing(n)} - X_{eoff}) + \dots \\
e^{A(t_{on} + t_{off(n)})}(X_{eoff} - X_{eon}) + e^{At_{off(n)}}(X_{eon} - X_{eoff}) + X_{eoff}
\end{aligned} \tag{3.14}$$

Again, since  $X_{landing} = [0 \ t_a]^T$ , the above two equations can be solved for the solution vector  $[t_{off} \ t_a]^T$ . In both "Long Thrust" and "Short Thrust" cases, once we obtain a solu-



**Figure 3-3.** Sample time history the hopper with the reduced states shown at each mode boundary.

tion for  $t_{off}$  and  $t_a$ , we can find the fixed point of the Poincare Map associated with the activation of thrust by the following relationship:

$$X_N = e^{At_{off(n)}} \left( - \begin{bmatrix} 0 \\ t_{a(n)} \end{bmatrix} - X_{eoff} \right) + X_{eoff} \quad (3.15)$$

### 3.2.3 Derivation of Jacobians for Local Stability

To find the multivariable Jacobians of the Poincare Maps derived above, we follow the approach described in the previous chapter of composing the Jacobian as the product of the individual Jacobians of the maps that transition the state from one mode boundary to the next. This allows us to "reuse" common terms between the two mode sequences of interest, without having to recalculate the overall derivative from the Poincare Maps. Figure 3-3

shows the maps for each mode. The reduced state variables at each mode boundary that are mapped are:

$$\tilde{X}_N = \begin{bmatrix} y_N \\ \dot{y}_N \end{bmatrix} \quad \tilde{X}_L = \begin{bmatrix} t_L \\ \dot{y}_L \end{bmatrix} \quad \tilde{X}_T = \begin{bmatrix} t_T \\ \dot{y}_T \end{bmatrix} \quad \tilde{X}_D = \begin{bmatrix} y_D \\ \dot{y}_D \end{bmatrix} \quad (3.16)$$

(As a reminder, upper case letters denote vectors, while lower case letters denote scalar variables. The tilda notation denotes the reduced state variable vector for that given mode boundary, as explained in Chapter 2). The overall Jacobian for both "Long Thrust" and "Short Thrust" for Poincare Maps about the instant that landing occurs can be written as the product of the individual Jacobians for each mode:

$$\text{Long Thrust:} \quad \frac{\partial \tilde{X}_{L(n+1)}}{\partial \tilde{X}_{L(n)}} = \frac{\partial \tilde{X}_L}{\partial \tilde{X}_T} \cdot \frac{\partial \tilde{X}_T}{\partial \tilde{X}_N} \cdot \frac{\partial \tilde{X}_N}{\partial \tilde{X}_L} \quad (3.17)$$

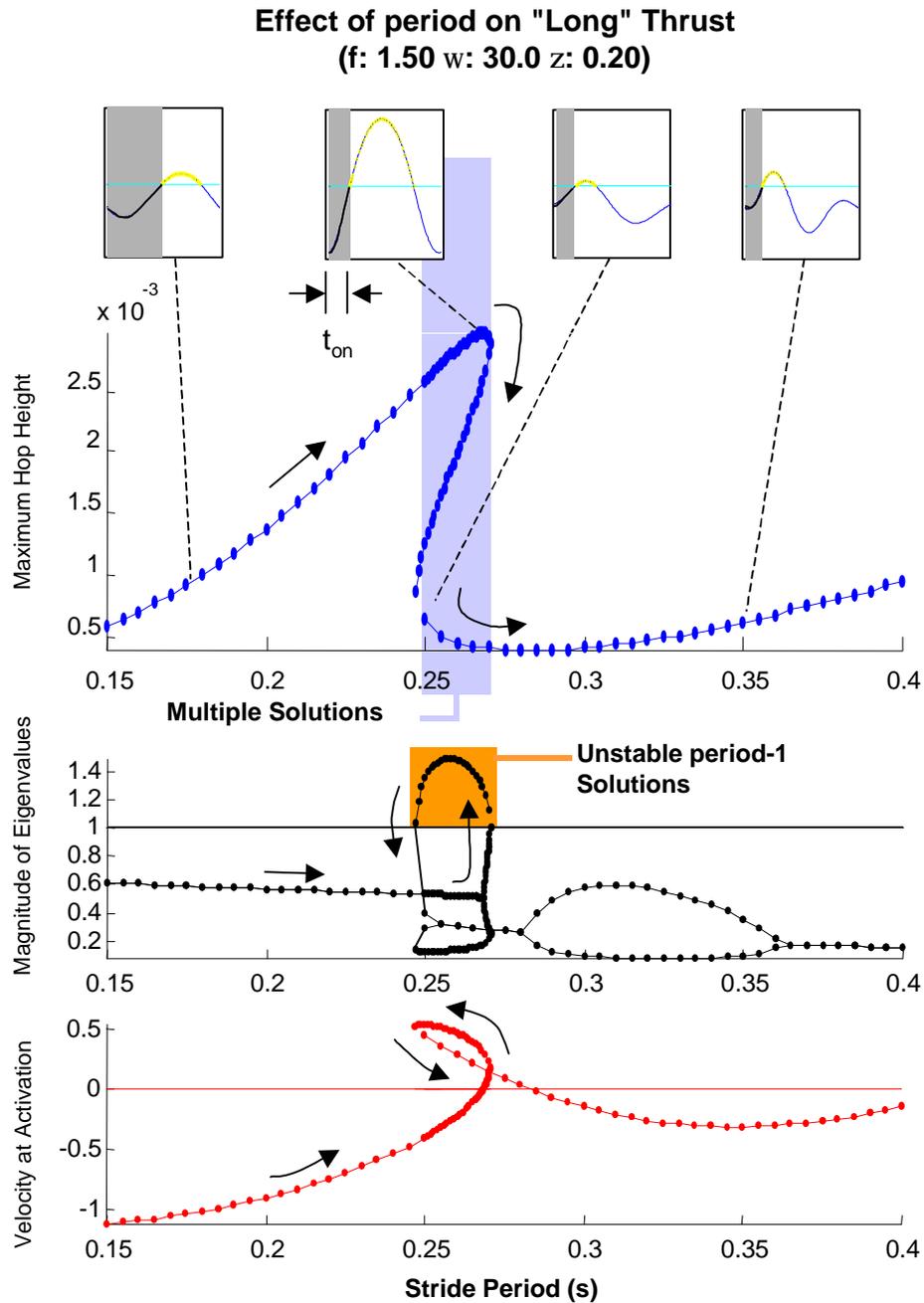
$$\text{Short Thrust:} \quad \frac{\partial \tilde{X}_{L(n+1)}}{\partial \tilde{X}_{L(n)}} = \frac{\partial \tilde{X}_L}{\partial \tilde{X}_T} \cdot \frac{\partial \tilde{X}_T}{\partial \tilde{X}_D} \cdot \frac{\partial \tilde{X}_D}{\partial \tilde{X}_N} \cdot \frac{\partial \tilde{X}_N}{\partial \tilde{X}_L} \quad (3.18)$$

Appendix B contains the derivation of these individual Jacobian matrices for the vertical hopper model. Having found each of the products in Equations 3.17 and 3.18, we can compose the overall Jacobians for the Poincare Maps for the two cases considered. As stated in the previous chapter, an advantage of this methodology is that we can formulate the Jacobian for Poincare Maps based on any particular event along the periodic trajectory by simply changing the order of the products of Equations 3.17 and 3.18.

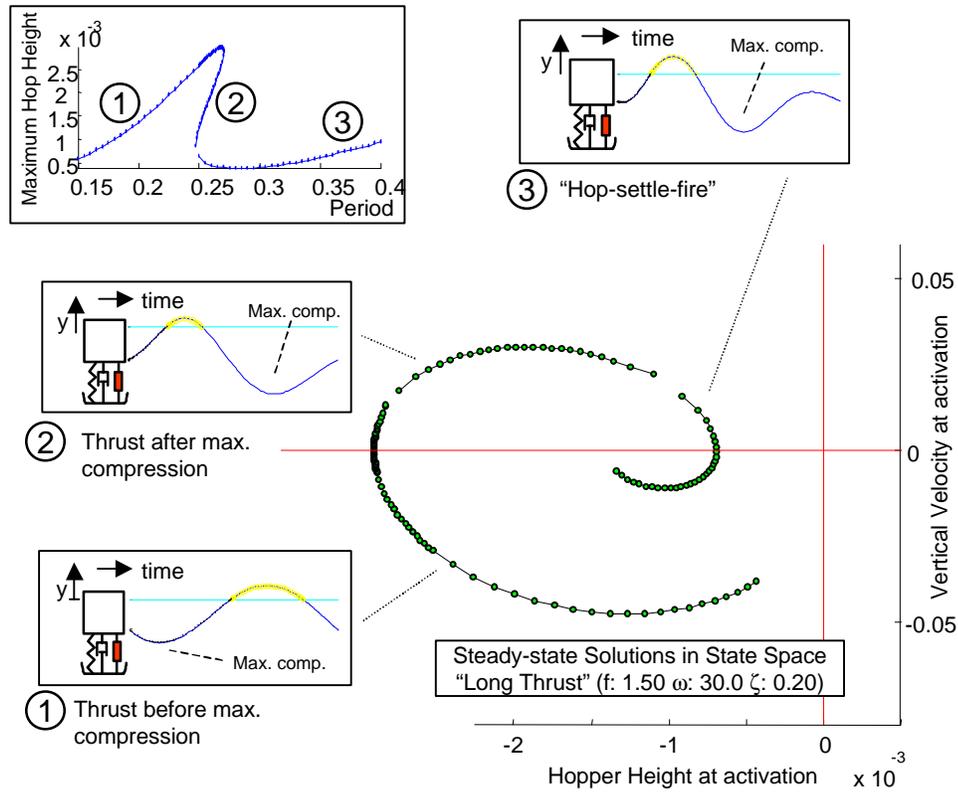
With these expressions for the Jacobians, we can determine the local stability of a particular periodic orbit (found numerically using the equations of the previous section) by evaluating the Jacobian at the steady-state solution and computing its eigenvalues.

### 3.2.4 Solution Search Results

Using the expressions found above for the Poincare Maps, steady-state solutions for a range of periods of the open-loop motor pattern were found numerically, given a set of prototypical system parameters, using the "fsolve" function in Matlab<sup>TM</sup> with a tolerance of  $10^{-8}$ .



**Figure 3-4.** Steady-state solutions as a function of stride period for the "Long Thrust" case. The arrows indicate how the continuum of solutions folds unto itself. The arrows also show how the continuum of the magnitude of the eigenvalues splits from two complex eigenvalues with the same magnitude to an unstable and a stable real eigenvalue.



**Figure 3-5.** Multiple steady-state solutions for a given stride period. The plot in the lower right shows the continuum of steady-state solutions plotted in the state-space of the state at thrust activation. The continuum is separated into three segments, each characterized by when thrust is initiated relative to maximum compression.

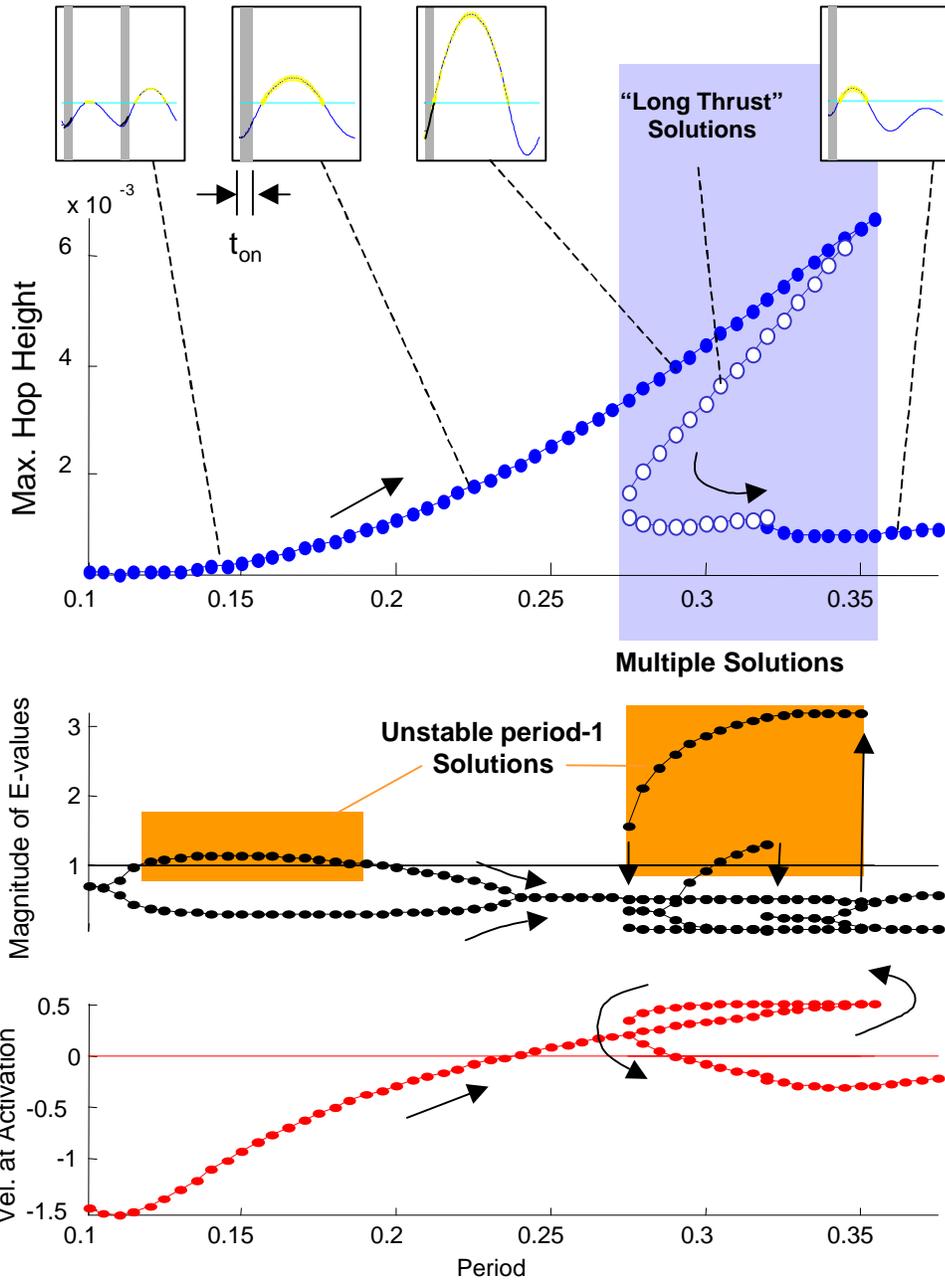
For each solution, the maximum hopping height, the velocity at thrust activation normalized by the velocity at take-off, and the magnitudes of the eigenvalues of the Jacobian evaluated at each solution are computed.

For the "Long Thrust" case, Figure 3-4 illustrates the effects of changing the open-loop stride period with a given normalized thrust magnitude,  $f=1.5$ , natural frequency,  $\omega=30$ , and damping ratio,  $\zeta=0.2$ . For short stride periods, hopping height starts out very small, as shown in the top plot. At these periods, thrust application starts well before maximum compression, given by the negative velocity at thrust application (in the figure, this velocity is normalized by the magnitude of the take-off velocity). These solutions are termed "Regular Hopping" as they represent a desired mode of hopping behavior. As the stride period is increased, hopping height increases, and velocity at application approaches zero. Finally, at a certain period (near 275ms period), height is maximized when velocity at thrust application is nearly zero (corresponding to thrust activation at maximum com-

pression), as predicted by previous studies. However, as the period approaches 275ms the magnitude of the eigenvalues quickly increases and the solution becomes unstable. Simulations of the hopper, though, never reach this point. As shown in the figure, other solutions to the state-steady conditions become available at a period near 250ms as the continuum of solutions folds back with respect to stride period. Of the two new sets of solutions available in this range of stride periods, one of them involves activating thrust after maximum compression and is unstable. The instability of these solutions may explain why they were never reported by previous studies (Ringrose, 1998 and Berkemeier and Desai, 1998), which found solutions using simulation. The solutions in the second set are termed “Hop-settle-fire” as the mass has started to settle before thrust is applied. These solutions are a result of including both significant damping and the effects of gravity. Previous studies which neglected both gravity and damping did not report these solutions. The hopping heights for these solutions are much lower, but their corresponding eigenvalues are also much lower, and the simulations converge to these solutions. Figure 3-5 illustrates these three sets of solutions. In the figure, the steady-state solutions, in the form of the state (height and velocity) at thrust activation, are plotted in state space in the lower right corner.

For the "Short Thrust" case, Figure 3-6 illustrates the effects of changing the open-loop stride period, given a fixed thrust duration  $t_{on}=0.02s$ . For short periods, the solutions start out as “Regular Hopping.” However, as the period is increased, the eigenvalues start to move outside the unit circle. Simulations for this range are “period-1 unstable” (the state does not repeat after one cycle), but tend to be “period-2 stable” (the state repeats after two cycles). As the period is further increased, the “Regular Hopping” solutions become period-1 stable again. The velocity at thrust activation also increases, and changes from negative (thrusting before maximum compression) to positive (thrusting after maximum compression). Maximum hopping height is also increased with period and keeps increasing until the continuum of "Short Thrust" solutions end. At this point, thrust deactivation starts to occur after take-off, making the solutions "Long Thrust." As discussed previously, these "Long Thrust" solutions in which thrust is activated after maximum compression are unstable, and "fold back" with respect to period. However, near 275ms period another continuum of valid steady-state solutions begins. These solutions are “Hop-settle-fire” solutions, for which hopping height is lower.

**Effect of period on "Short" Thrust**  
 (f: 3.20 w: 30.0 z: 0.20  $t_{on}$ : 0.02)



**Figure 3-6.** Steady-state solutions as a function of stride period for the "Short Thrust" case. Arrows indicate how the continuum of solutions folds onto itself. At some point, thrust deactivation starts to occur after take-off, and the continuum switches to "Long Thrust" solutions.

A rigorous analysis of the effects of the model's other parameters on the hopping motion is beyond the scope of this work, but they are nonetheless described here from experience with the simulations. The onset of the "Hop-settle-fire" solutions is determined

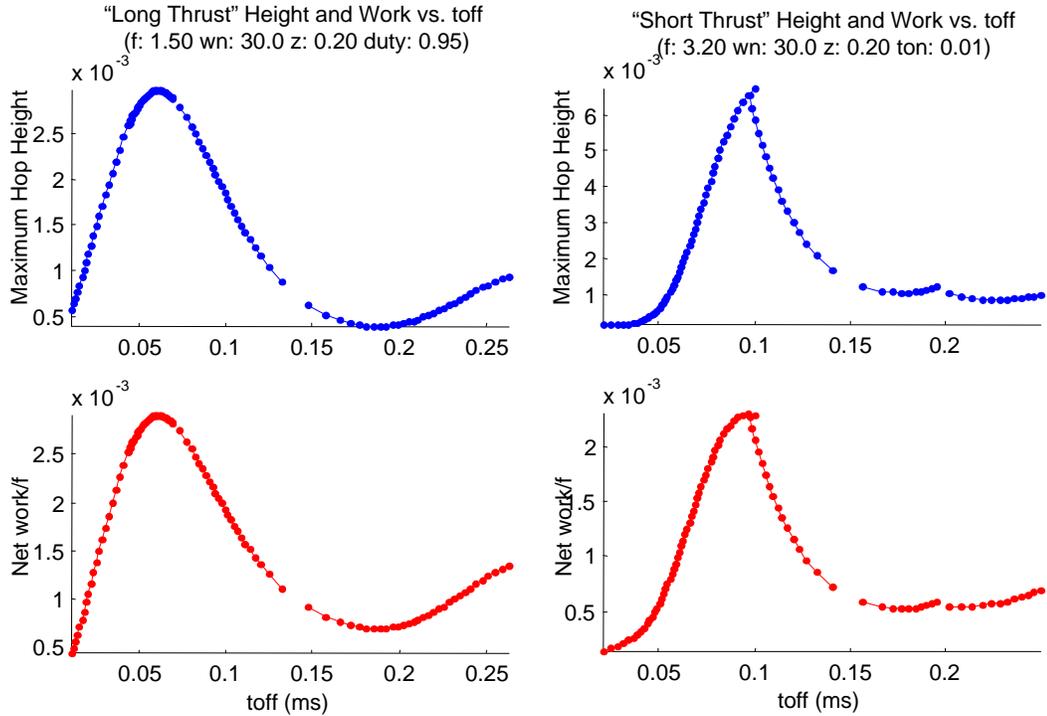
largely by the system's natural frequency. These solutions become available when the period of thrust application is long enough that the system is allowed to settle according to its natural period. The addition of damping also makes these solutions possible, since without damping the system would not settle. Decreasing damping and increasing the thrust magnitude and duration all seem to have a destabilizing effect, as they extend the region of unstable solutions.

As shown in these sample results, performance in terms of hopping height varies significantly with the period of the motor pattern. In addition, multiple solutions exist for a range of periods, and certain solutions become unstable over other ranges of periods. The following sections help explain these behaviors and seek to characterize the factors that determine performance and stability.

### 3.3 Characterizing Performance

Performance in a vertical hopper in terms of maximum hopping height can be characterized by the net amount of work done by the forcing function within one hop. This is explained by the following statements. A given steady-state solution is such that the total amount of energy does not change within the stride period (the total amount of energy injected by the forcing function equals the total amount of energy passively dissipated by damping in the system). Depending on the relative phasing of the forcing function and the motion of the system, the forcing function may perform both positive and negative work. Steady-state orbits in which the forcing function performs more work will entail higher hopping heights, since the energy that is necessarily dissipated through damping is proportional to the square of the landing velocity (as observed by Ringrose, 1998), and hopping height is proportional to the square root of the landing velocity. As shown in Figure 3-7, for both "Long Thrust" and "Short Thrust" cases, the hopping height is nearly proportional to the work performed by the actuator.

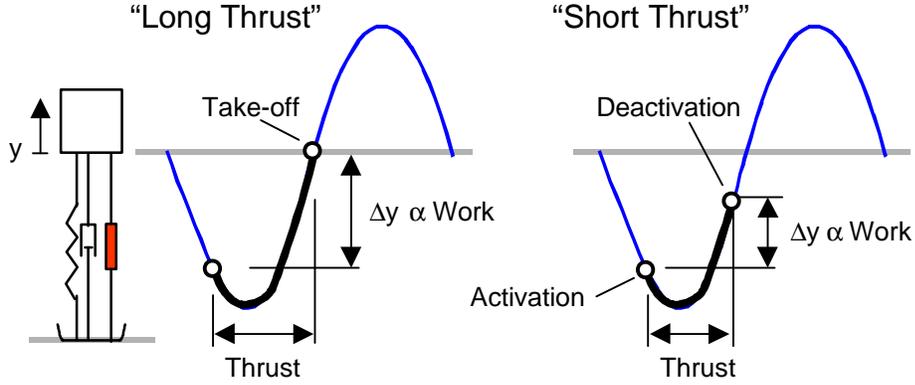
The quantities of work and hopping height in Figure 3-7 are plotted against  $t_{off}$ , the time after landing that thrust is activated. As shown, there are no multiple solutions with respect to  $t_{off}$ . Multiple solutions arise when plotting against period due to the fact that the airborne time duration is proportional to the take-off velocity. Assuming that the stance



**Figure 3-7.** Height and work versus  $t_{off}$ , the time after landing that thrust is initiated. As shown, solutions in which maximum work is performed correspond to maximum height. Note also that multiple solutions do not exist when plotting against  $t_{off}$ .

time does not vary as much as the airborne time, the total stride period peaks for trajectories with higher hopping height. As a result, solutions that move away from this peak can have the same stride period. This is one of the fundamental limitations of open-loop control for vertical hopping, since only the period of the motor pattern can be varied. Thus, control is lost over which solution the system will converge to for periods of the motor pattern in this range. While experience with the model suggests that increasing damping and lowering the thrust magnitude help decrease the occurrence of multiple solutions, it is worth noting that using  $t_{off}$  as a control parameter (for example, thrust could be activated by a timer that is started when the foot contacts the ground) avoids multiple solutions, but at the expense of added dependence on sensory information.

We can further characterize the work performed by the forcing function, and thus characterize performance, by looking at the state at thrust activation and deactivation relative to the motion of the system. The work performed by the forcing function during one stride is equal to the integral of the force times the velocity of the mass:



**Figure 3-8.** Work performed by the actuator in this model is proportional to the height difference between the state at thrust activation and thrust deactivation.

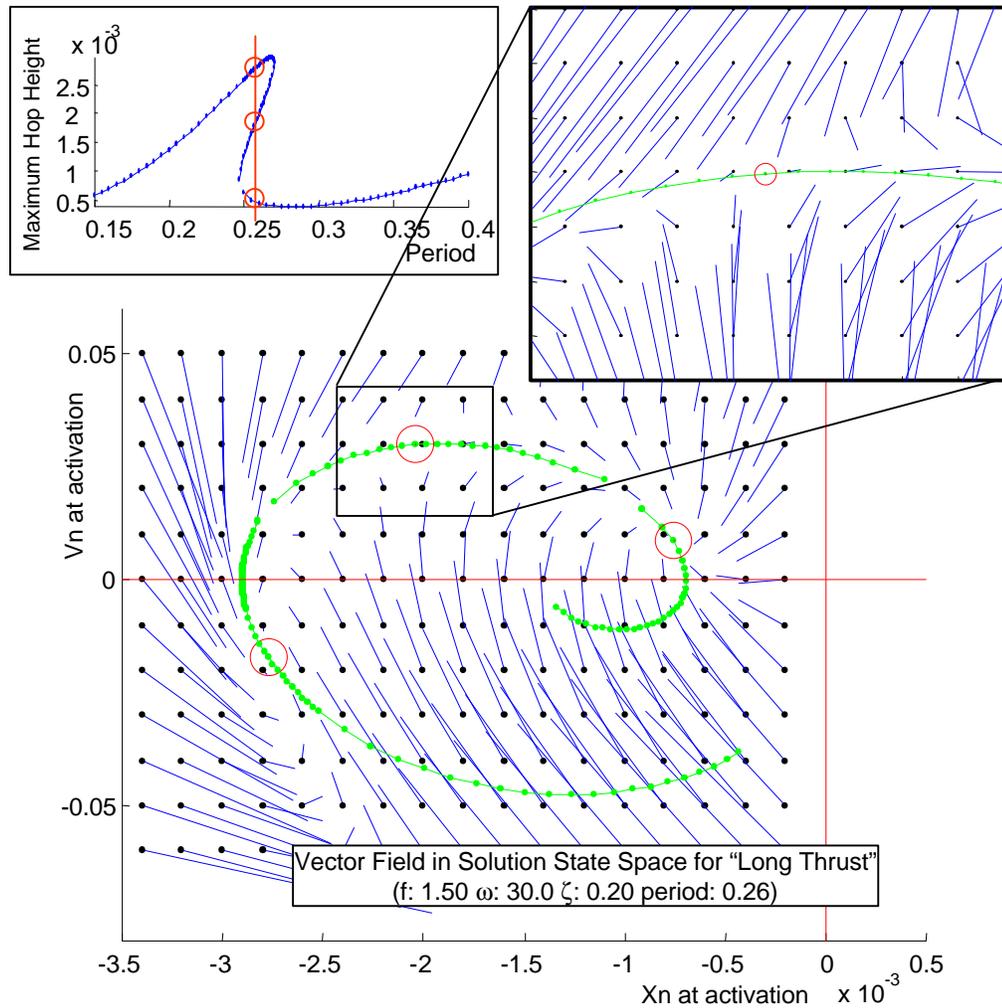
$$\int_t^{(t+\tau)} f(t) \cdot \dot{y}(t) dt = \text{WorkInput} \quad (3.19)$$

If  $f(t)$  is a square wave of magnitude  $f$  with time duration  $t_{on}$ , this integral becomes:

$$f \cdot \int_{t_{off}}^{(t_{off}+t_{on})} \dot{y}(t) dt = f \cdot (y(t_{off}+t_{on}) - y(t_{off})) \quad (3.20)$$

Work is thus proportional to the difference in height between thrust deactivation and thrust activation. For "Long Thrust" situations, this is equivalent to maximizing hopping height by initiating thrust at maximum compression, which correlates with the results of previous researchers. This is illustrated in Figure 3-8. For "Short Thrust" situations, however, this is not the case. For a given thrust duration, work is maximized by placing thrust deactivation near take-off, which may put thrust activation after maximum compression. This is also illustrated in Figure 3-8. As a result, hoppers such as Raibert's, in which thrust is initiated closed-loop when the leg is sensed to be at maximum compression and deactivated after a certain period of time, may be operating suboptimally.

These results give indication that performance can be monitored and evaluated by measuring when thrust is activated and deactivated relative to take-off and maximum compression, which require simple sensing compared to the task of full state estimation. This opens the possibility of adapting, or tuning, the stride period, or other parameters of the open-loop motor pattern, with minimal, or low-bandwidth, sensing.



**Figure 3-9.** Unstable steady-state solution for "Long Thrust" case. At a period of 0.26, there are three valid steady-state solutions. The solution in which thrust is activated shortly after maximum compression is unstable, as illustrated by the vector field. In the vector field, the dots indicate the state at the time that thrust is initiated. The vector lines indicate the direction from this initial condition towards the state after one period of the motor pattern.

Although we've begun to characterize performance for these steady-state orbits, there are still the questions of whether a given trajectory is stable or unstable and why in simulation the system prefers some solutions more than others. These are addressed in the following section.

### 3.4 Characterizing Stability

Examination of the plots of the eigenvalues in Figure 3-4 and Figure 3-6 indicates that stability properties of the steady-state orbits are different in the "Long Thrust" and "Short Thrust" cases. In the "Long Thrust" case, instability occurs for a range of solutions in which

thrust is activated shortly after maximum compression, and before the solutions become "Hop-settle-fire." In the "Short Thrust" case, instability occurs for a range of solutions in which thrust is activated before maximum compression, take-off velocities are low and the stride period is small. In the following, these observations are validated using the analytical expressions for the Jacobians previously found, and physical intuition for the unstable behavior is provided.

### **3.4.1 Stability for "Long Thrust" Case**

As noted, unstable solutions for the "Long Thrust" case initiate thrust shortly after maximum compression. These solutions share the same range of stride periods as a set of solutions in which thrust is initiated before maximum compression and a set of "Hop-settle-fire" solutions (see Figure 3-4). In order to visualize what is happening for one such period in this range of multiple solutions, Figure 3-9 shows a vector field in the state space of solutions for the Poincare Map formulated about the state at thrust activation. The spiral curve represents the solutions presented earlier. The circles are the three simultaneous solutions for a period of 0.26s. To form the vector field, the model was simulated starting at each point in a grid in the vector space, at the instant of thrust activation. After one period of 0.26s, a vector line was drawn between the initial point and the state at the end of the simulation, here scaled by 0.4. Thus, the vector field represents the "flow" of the state sampled discretely at the instant of thrust initiation. This figure helps explain several observed behaviors. First, near periods in which the three solutions start to appear, at the edge of the range of multiple solutions, simulations exhibit chaotic, or non-periodic motion. From the figure, one can see how simulations may oscillate non-periodically between the domains of attraction of the two stable solutions. Second, the domain of attraction of the "Hop-settle-fire" solution (right-most circle in the vector field) appears to be larger than the domain of the solution in which thrust is initiated before maximum compression (left-most circle). Although both solutions are stable, simulations with periods well into the range of multiple solutions tend to converge to "Hop-settle-fire."

From the figure, we can also see that the unstable solutions in which thrust is initiated shortly after maximum compression are "saddle-points," that is, they appear to have

one stable mode and one unstable mode. Looking at the Jacobian of the Poincare Map, we can show that these solutions are necessarily unstable.

The Jacobian presented earlier used  $X_T=[y_T \dot{y}_T]^T$  in its derivation, which greatly simplified the resulting expressions. In this case, however, it is more useful to keep  $X_N$ , the state at thrust activation, in the expressions. To do this, we rederive the Jacobian from the Poincare Map based on the state at thrust activation, presented earlier in Equation 3.8, here repeated:

$$X_{N(n+1)} = F(X_n) = X_{eoff} - e^{A(\tau - 2t_{a(n)} - t_{on(n)})} (X_{eon} + X_{eoff}) \dots - e^{A(\tau - 2t_{a(n)})} (X_{N(n)} - X_{eon}) \quad (3.21)$$

Taking the derivative with respect to  $X_{N(n)}$ , we get:

$$\left. \frac{\partial X_{N(n+1)}}{\partial X_{N(n)}} \right|_{X_N^*} = 2A(X_{eoff} - X_N^*) \left. \frac{\partial t_a}{\partial X_{N(n)}} \right|_{X_N^*} + \dots - e^{A(\tau^* - 2t_a^*)} A e^{At_{off}^*} (X_{eon} + X_{eoff}) \left. \frac{\partial t_{on}}{\partial X_{N(n)}} \right|_{X_N^*} \quad (3.22)$$

The asterix notation in this case denotes the fixed point. In order to find the expressions  $\partial t_a / (\partial X_{N(n)})$  and  $\partial t_{on} / (\partial X_{N(n)})$ , we use the take-off condition:

$$\begin{bmatrix} 0 \\ t_a \end{bmatrix} = e^{At_{on}} (X_{N(n)} - X_{eon}) + X_{eon} \quad (3.23)$$

and take derivatives with respect to  $X_{N(n)}$  to find:

$$\begin{bmatrix} 0 & 0 \\ \partial t_a / \partial X_{N(n)} \end{bmatrix} = A e^{At_{on}} (X_{N(n)} - X_{eon}) \frac{\partial t_{on}}{\partial X_n} + e^{At_{on}} \quad (3.24)$$

Decomposing each row and rearranging to solve for  $\partial t_a / (\partial X_{N(n)})$  and  $\partial t_{on} / (\partial X_{N(n)})$  yields,

$$\left. \frac{\partial t_{on}}{\partial X_{N(n)}} \right|_{X^*} = - \left[ \begin{bmatrix} 1 & 0 \end{bmatrix} A e^{At_{on}^*} (X_N^* - X_{eon}) \right]^{-1} \begin{bmatrix} 1 & 0 \end{bmatrix} e^{At_{on}^*} \quad (3.25)$$

$$\left. \frac{\partial t_a}{\partial X_{N(n)}} \right|_{X^*} = \begin{bmatrix} 0 & 1 \end{bmatrix} e^{At_{on}^*} \left[ A(X_{N^*} - X_{eon}) \left. \frac{\partial t_{on}}{\partial X_{N(n)}} \right|_{X^*} + I \right] \quad (3.26)$$

To analyze the stability of this Jacobian, we use the Jury Stability Test (Ogata, 1985), which states that a discrete system of two dimensions with the characteristic equation  $P(z)$  of the form:

$$P(z) = a_0 z^2 + a_1 z + a_2 \quad (3.27)$$

where  $a_0 > 0$ , is stable if the following conditions are all satisfied:

$$\begin{aligned} |a_2| &< a_0 \\ a_0 + a_1 + a_2 &> 0 \\ a_0 - a_1 + a_2 &> 0 \\ |(a_0 + a_2) \cdot (a_2 - a_0)| &> |a_1 \cdot (a_0 - a_1)| \end{aligned} \quad (3.28)$$

For a Jacobian of the form:

$$\left. \frac{\partial X_{N(n+1)}}{\partial X_{N(n)}} \right|_{X_{N^*}} = \begin{bmatrix} J_1 & J_2 \\ J_3 & J_4 \end{bmatrix} \quad (3.29)$$

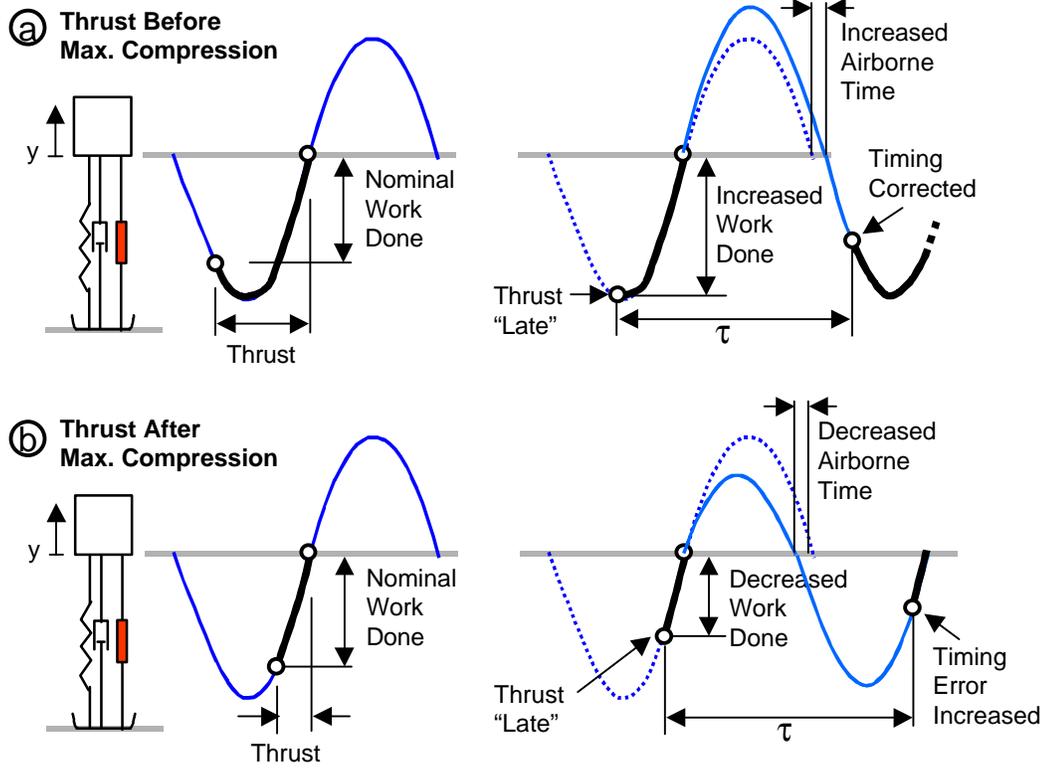
we have,

$$P(z) = z^2 - (J_1 + J_4)z + (J_1 J_4 - J_2 J_3) \quad (3.30)$$

Substituting into the conditions in Equation 3.28,

$$\begin{aligned} |(J_1 J_4 - J_2 J_3)| &< 1 \\ 1 - (J_1 + J_4) + (J_1 J_4 - J_2 J_3) &> 0 \\ 1 + (J_1 + J_4) + (J_1 J_4 - J_2 J_3) &> 0 \\ |(1 + (J_1 J_4 - J_2 J_3)) \cdot ((J_1 J_4 - J_2 J_3) - 1)| &> |(J_1 + J_4) \cdot (1 + (J_1 + J_4))| \end{aligned} \quad (3.31)$$

Of these conditions, the one that plays a significant role is the third condition. Substituting the elements of the Jacobian found in Equations 3.21, 3.25 and 3.26 into the third



**Figure 3-10.** Effect of perturbations in the timing of thrust activation on the work performed, and resulting take-off velocity and airborne phase duration for the "Long Thrust" case.

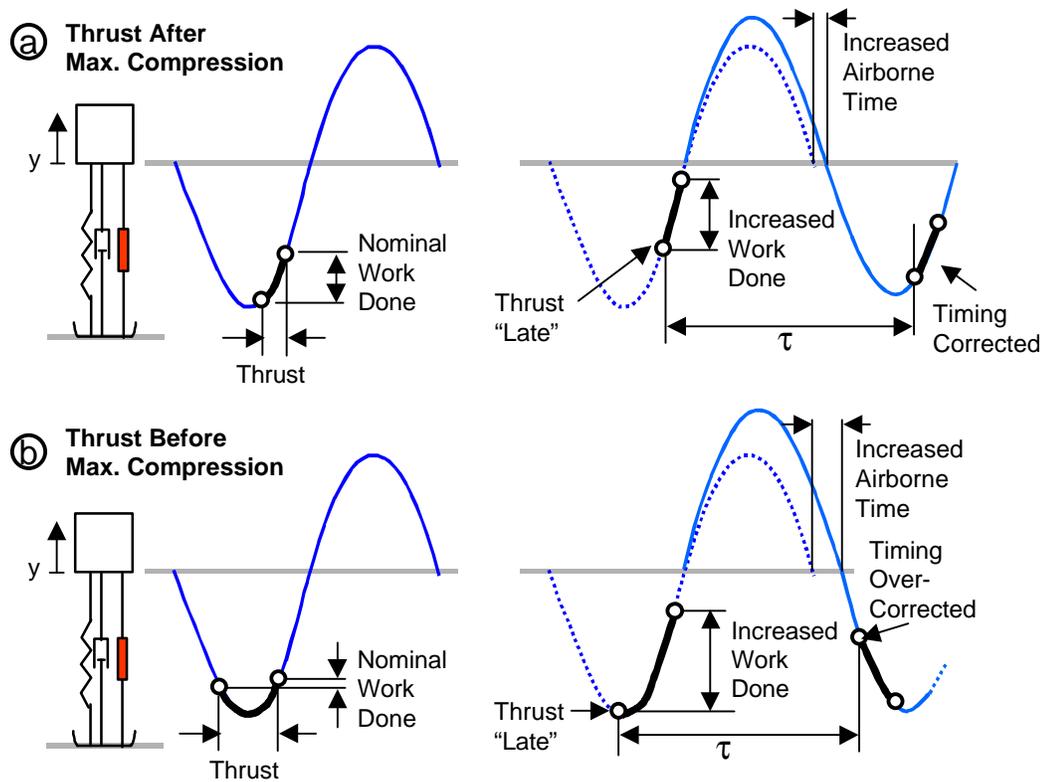
condition, expanding all the terms, and simplifying using Matlab's symbolic toolbox (The Mathworks, Inc.), the third condition becomes:

$$e^{-\zeta\omega_n t_{on}} \left( \frac{f}{\omega_d t_d} \right) (\sin(\omega_d t_{on}) - e^{-\zeta\omega_n(\tau - 2t_a)} \sin(\omega_d t_{off}) - 2\omega_d e^{-\zeta\omega_n t_{on}} \dot{y}_N) > 0 \quad (3.32)$$

where  $\omega_d$  is the damped natural frequency of the mass-spring-damper system, and  $\dot{y}_N$  is the velocity at thrust activation. Since the first two products of the expression are always positive, the condition reduces to:

$$\sin(\omega_d t_{on}) - e^{-\zeta\omega_n(\tau - 2t_a)} \sin(\omega_d t_{off}) - 2\omega_d e^{-\zeta\omega_n t_{on}} \dot{y}_N > 0 \quad (3.33)$$

From this expression, we can see analytically that stability depends largely on the sign of  $\dot{y}_N$ . For solutions in which thrust occurs after maximum compression,  $\dot{y}_N$  is positive, which makes the third term in the expression negative. Since  $t_{on}$  is approximately

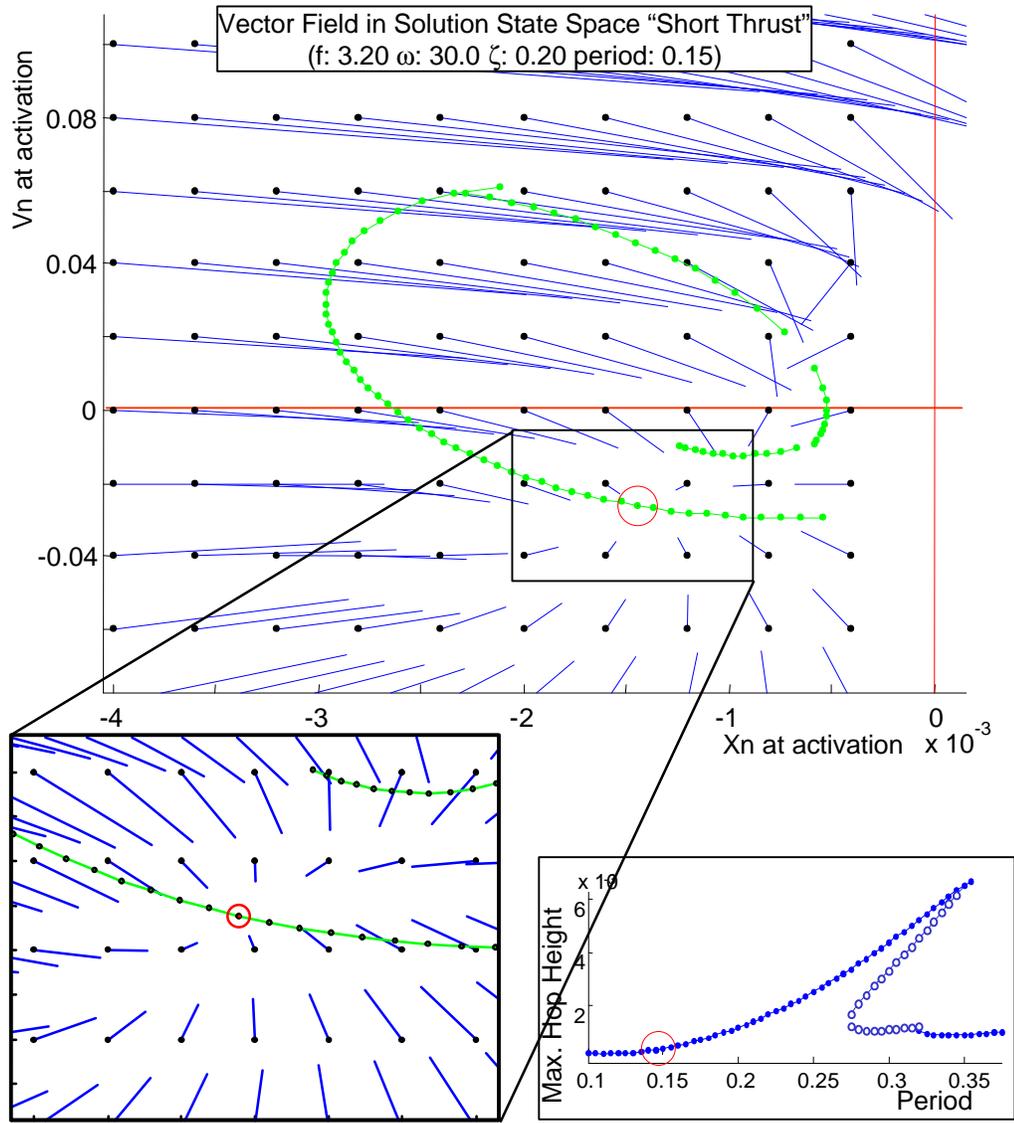


**Figure 3-11.** Effect of perturbations in the timing of thrust activation on the work performed, and resulting take-off velocity and airborne phase duration for the "Short Thrust" case.

equal to  $t_{off}$  for these solutions, this third term, and thus the sign of  $\dot{y}_N$ , plays a large role in determining stability, regardless of thrust magnitude or other system parameters.

Intuitively, this can be explained by looking at the effect on work input that perturbations about steady-state have for solutions in which thrust is activated after maximum compression. As illustrated in Figure 3-10a, thrusting "late," when it occurs before maximum compression, results in an increase in the work performed by the actuator. This in turn results in a higher take-off velocity, and a longer airborne phase. Since the stride period is fixed by the motor pattern, this elongated air phase means that thrust initiation will occur earlier relative to landing, thereby correcting for the initial disturbance. When thrust is initiated after maximum compression, thrusting "late" causes a decrease in work, a subsequent shorter airborne phase, and an even "later" thrust initiation in the next cycle, resulting in unstable behavior, as shown in Figure 3-10b.

Although this mechanism for the correction of timing of thrust initiation works to stabilize the case of "Long Thrust" when thrust is activated before maximum compression,



**Figure 3-12.** Unstable steady-state solution for "Short Thrust" case for a period of 0.15. At this period, thrust is initiated well before maximum compression, and take-off velocity is small.

it could also possibly "over-correct" the timing, resulting in unstable behavior. This has not been observed in the "Long Thrust" case, but it is a source of instability for the "Short Thrust" case, as shown in the following section.

### 3.4.2 Stability for "Short Thrust" case

In the "Short Thrust" case, solutions in which thrust is initiated after maximum compression are not necessarily unstable as in the "Long Thrust" case. In this case, initiating thrust "late" moves both thrust activation and deactivation later, which increases the work done,

as illustrated in Figure 3-11a, causing a longer airborne phase and a correction to the error in thrust timing.

As shown previously in Figure 3-6, instability for the "Short Thrust" case occurs in two ranges of periods. The first is in short periods for which thrust activation occurs well before maximum compression and hopping height is low. The second is when period is increased such that thrust activation occurs after maximum compression and thrust deactivation occurs after take-off, in which case the system becomes "Long Thrust" and is unstable as explained in the previous section. The first case is of interest, and is visualized in Figure 3-12. This figure shows a vector field in the solution state space similar to the one presented in Figure 3-9, but for the "Short Thrust" solutions, and for a period in which the eigenvalues are unstable. Although the vectors are directed towards the steady-state solution, its eigenvalues have a magnitude greater than one. As shown next, a look at the analytical expression of the Jacobian for this range of periods helps explain the nature of this instability.

The Jacobian for the "Short Thrust" case is found by substituting the expressions found in Equations B.9, B.11, B.23 and B.34 in Appendix B into the expression for the Jacobian in Equation 3.18, yielding:

$$\frac{\partial \tilde{X}_{L(n+1)}}{\partial \tilde{X}_{L(n)}} = \frac{\partial \tilde{X}_L}{\partial \tilde{X}_T} \cdot \frac{\partial \tilde{X}_T}{\partial \tilde{X}_D} \cdot \frac{\partial \tilde{X}_D}{\partial \tilde{X}_N} \cdot \frac{\partial \tilde{X}_N}{\partial \tilde{X}_L} = \dots \quad (3.34)$$

$$-\frac{1}{\dot{y}_T} \begin{bmatrix} 1 & 2 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ (-1 - 2\zeta\omega\dot{y}_T) & -\dot{y}_T \end{bmatrix} e^{At_{off2}} e^{At_{on}} e^{At_{off}} \begin{bmatrix} \dot{y}_T & 0 \\ (1 - 2\zeta\omega\dot{y}_T) & 1 \end{bmatrix}$$

We can further simplify by noting that  $t_{off2} + t_{on} + t_{off}$  is equal to the total stance time,  $t_{stance}$ :

$$\frac{\partial \tilde{X}_{L(n+1)}}{\partial \tilde{X}_{L(n)}} = -\frac{1}{\dot{y}_T} \begin{bmatrix} 1 & 2 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ (-1 - 2\zeta\omega\dot{y}_T) & -\dot{y}_T \end{bmatrix} e^{At_{stance}} \begin{bmatrix} \dot{y}_T & 0 \\ (1 - 2\zeta\omega\dot{y}_T) & 1 \end{bmatrix} \quad (3.35)$$

From this expression, we can see what can happen for solutions in which thrust is activated well before maximum compression. For these solutions,  $\dot{y}_T$  is small (but positive, since it is the velocity at take-off) and the stance time,  $t_{stance}$ , is approximately one half of

the mass-spring-damper system's natural period (assume 45% of the natural period). Using this approximation,

$$t_{\text{stance}} \approx 0.45\tau_n \quad (3.36)$$

where  $\tau_n$  is the mass-spring-damper system's natural period,  $e^{At_{\text{stance}}}$  can be approximated as follows:

$$e^{At_{\text{stance}}} = \begin{bmatrix} c(\omega_d t_{\text{stance}}) + \zeta\omega_n s(\omega_d t_{\text{stance}})/\omega_d & s(\omega_d t_{\text{stance}})/\omega_d \\ -\omega_n^2 s(\omega_d t_{\text{stance}})/\omega_d & c(\omega_d t_{\text{stance}}) - \zeta\omega_n s(\omega_d t_{\text{stance}})/\omega_d \end{bmatrix} \quad (3.37)$$

$$e^{At_{\text{stance}}} \approx e^{-\zeta\omega_n t_{\text{stance}}} \begin{bmatrix} -1 & 0 \\ -b & -1 \end{bmatrix}$$

where  $s$  and  $c$  denote the sine and cosine functions, and  $b$  is a number equal to:

$$b = \omega_n^2 s(\omega_d t_{\text{stance}})/\omega_d \quad (3.38)$$

the expression for the Jacobian then becomes:

$$\begin{aligned} \frac{\partial \tilde{X}_{L(n+1)}}{\partial \tilde{X}_{L(n)}} &\approx -\frac{1}{\dot{y}_T} e^{-\zeta\omega_n t_{\text{stance}}} \begin{bmatrix} 1 & 2 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} 1 & 0 \\ (-1-a) & \dot{y}_T \end{bmatrix} \begin{bmatrix} -1 & 0 \\ -b & -1 \end{bmatrix} \begin{bmatrix} \dot{y}_T & 0 \\ (1-a) & 1 \end{bmatrix} \\ &= -\frac{1}{\dot{y}_T} e^{-\zeta\omega_n t_{\text{stance}}} \begin{bmatrix} (1+2a+2\dot{y}_T b)\dot{y}_T + 2\dot{y}_T(1-a) & 2\dot{y}_T \\ (-1-a-\dot{y}_T b)\dot{y}_T - \dot{y}_T(1-a) & -\dot{y}_T \end{bmatrix} \end{aligned} \quad (3.39)$$

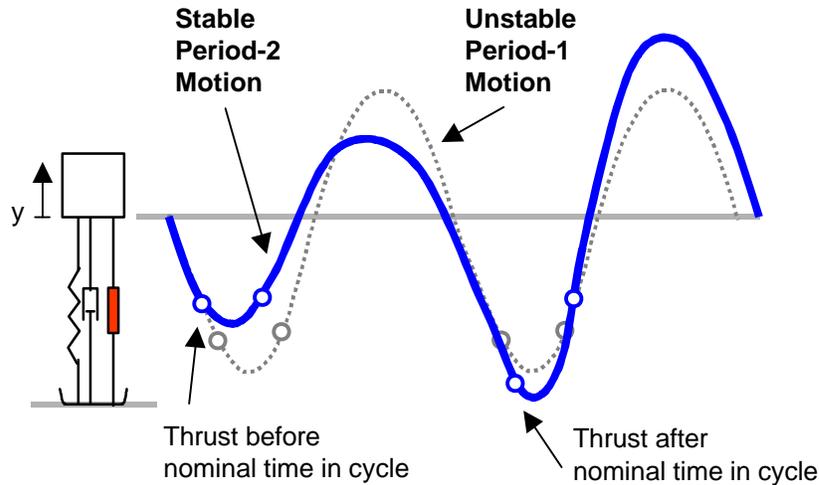
where,

$$a = 2\zeta\omega\dot{y}_T \quad (3.40)$$

The eigenvalues of this Jacobian are:

$$-e^{-\zeta\omega_n t_{\text{stance}}} \left( (1 + \dot{y}_T b) \pm \sqrt{(1 + \dot{y}_T b)^2 - 1} \right) \quad (3.41)$$

From this simple expression of the eigenvalues, we can see that both eigenvalues will be real and negative if  $\dot{y}_T b$  is small ( $\dot{y}_T b$  is approximately 0.2 for the case shown in Figure 3-12). If  $\dot{y}_T$  and  $t_{\text{stance}}$  are such that the larger eigenvalue is unstable, it will be



**Figure 3-13.** Stride periods for which unstable period-1 trajectories exist in the short thrust case tend to result in period-2 stable trajectories. These period-2 orbits alternately initiate thrust before and after the nominal point in the cycle where the unstable period-1 orbit initiates thrust. This behavior is explained by the negative sign of the eigenvalues in this regime, which indicate that perturbations about the nominal trajectory will be "overcompensated", causing thrust activation to oscillate about the unstable equilibrium.

because it is less than  $-1$ . We also observe from this result that increasing the damping ratio has the effect of stabilizing this eigenvalue. A negative eigenvalue less than  $-1$  implies that the system "over-compensates" or "over-corrects" for disturbances. This is illustrated in Figure 3-11b. In this situation, thrusting "late" will move both thrust activation and deactivation further along the cycle, decreasing the amount of negative work performed and increasing the amount of positive work. This large increase in the net work can cause a very large increase in the take-off velocity and ensuing airborne phase such that thrust occurs "very early" from the nominal timing, over-compensating for the initial disturbance. This nature of the instability in this range of periods helps explain why simulations in this range tend to be period-2 trajectories. These period-2 trajectories repeat themselves after two stride cycles, and tend to "oscillate" about the unstable period-1 trajectory, as illustrated in Figure 3-13. The negative sign of the eigenvalue indicates that perturbations in the time that thrust is activated along the period-1 trajectory will "overshoot" at the next cycle, causing thrust activation to occur alternately before and after the nominal point in the period-1 cycle.

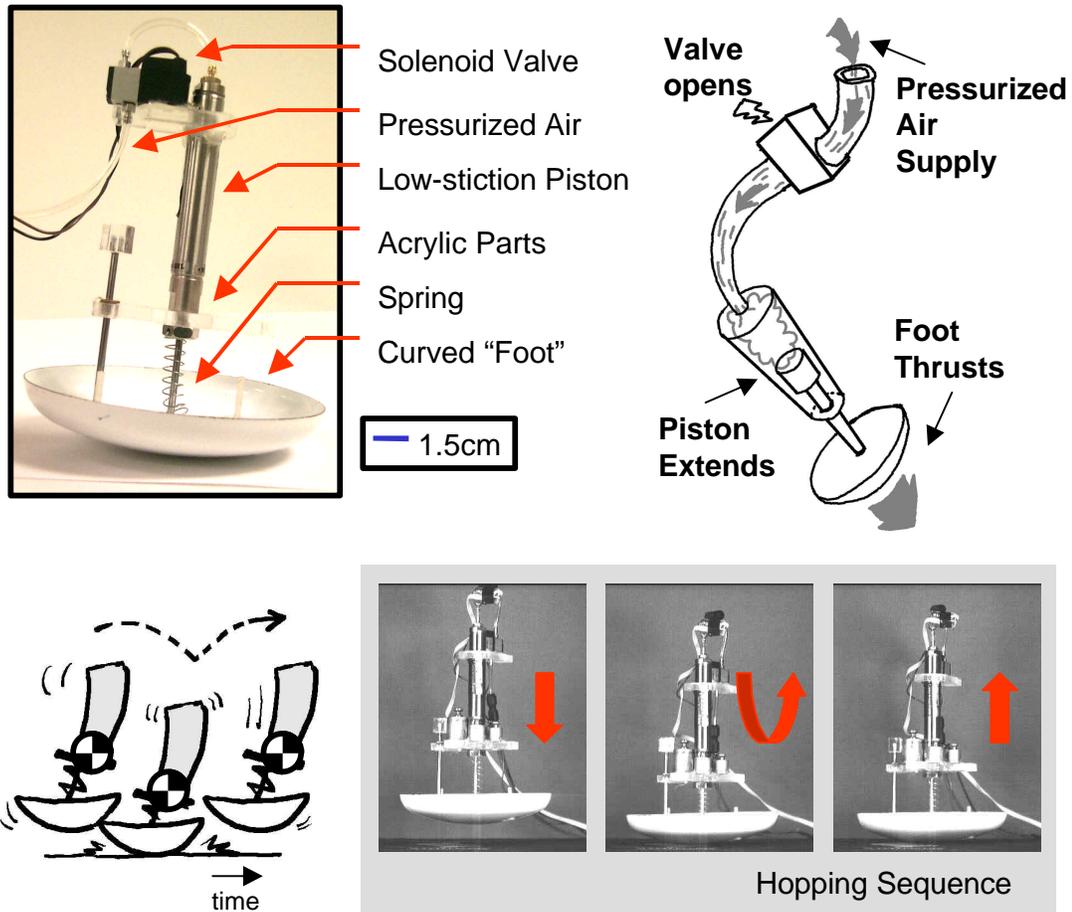


Figure 3-14. The "Dashpod," a pneumatically-actuated vertical hopper.

### 3.5 Experimental Verification - Dashpod Data

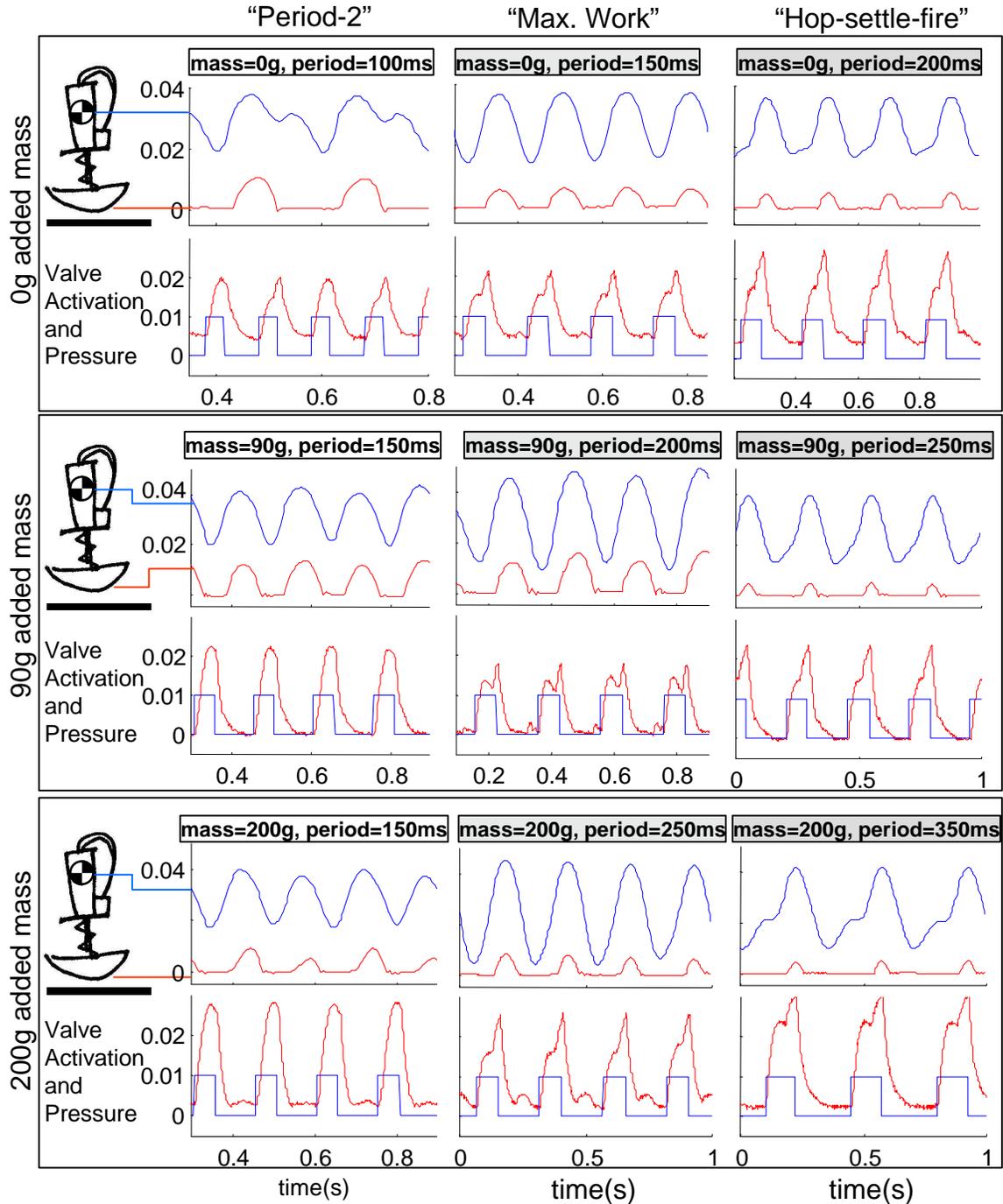
The characterizations presented in previous sections were based on simulation and analytical results for a simplified hopping model. In order to verify these observations, this section presents experimental data for a vertical hopping machine called the "Dashpod." The Dashpod is illustrated in Figure 3-14, and consists of a low-stiction pneumatic piston attached to a wide dish or curved "foot" on which it stands. A spring connects the foot to the pneumatic cylinder and platform along the piston shaft and a solenoid valve regulates pressurized air into the cylinder's upper chamber. The machine can be made to hop by activating the solenoid valve, which fills the cylinder's upper chamber with pressurized air, causing the piston to thrust against the ground, and pushing the Dashpod's platform up. The hopper maintains vertical orientation through the curved foot, which acts to stabilize the

machine when in contact with the ground. More details on the machine can be found in (Cham et al., 2001), which describes the use of the hopper for a set of educational laboratories in dynamic systems.

This machine is significantly more complex than the model used in the above analysis. Besides non-linearities in system parameters like stiffness, damping and friction, the foot of the machine has significant mass, which means that a certain amount of energy is used at take-off to lift the foot off the ground. This energy is lost when the foot impacts the ground at landing. Also, although the motor commands sent to the valve consist of a square wave, as in the model, the thrust force is proportional to the pressure inside the cylinder. This results in a delay in the application of thrust (the air dynamics can be modeled as a first order system, as shown in Cham et al., 2001), and in additional compliance and damping due to the compressibility of air. Despite these non-linearities, the Dashpod exhibits many of the behaviors predicted by the simplified model.

Figure 3-15 shows experimental data of the Dashpod while hopping. The data were taken through hi-speed video capture of reflective markers placed on the machine's body and foot. Air pressure at the cylinder was also recorded. The figure shows nine experimental trials, in which the mass of the body and the stride period were varied. For each trial, the top plot shows the time history of the height of the body and foot for several consecutive stride periods. The bottom plot shows the corresponding square wave motor command sent to the valve and the measured pressure at the cylinder.

In the figure, the left-most column of trials represents hopping trajectories in which the stride period is short compared to the dynamics of the system. As shown, thrust activation occurs before maximum compression, and thrust deactivation occurs shortly after. More interestingly, close examination of the hopping trajectories shows that the motion is period-2. This is very evident in the top-most trial, in which no extra mass was added to the Dashpod's body. The effect is more subtle in the other trials with added mass, but still evident in the differences in height of the body and foot trajectories between strides. This validates the observations made for the "Short Thrust" case in simulation and analytically in which period-2 behavior appears as a possible consequence of the "over-correcting" effect



**Figure 3-15.** Experimental data of hopping trajectories for the Dashpod vertical hopper. These nine trials represent variations in the hopper's mass and the period of the motor pattern.

present when thrust activation occurs before maximum compression for short stride periods.

The center column of trials represents hopping trajectories in which the hopper is activated with a period that is more "in tune" with the dynamics of the hopper. As shown,

thrust activation occurs near maximum compression of the spring, and deactivation occurs near lift-off, maximizing the work performed by the actuator, as evident in the high oscillations of the hopper body. Also of interest is the indication that these trajectories may be on the border of instability. As shown, the body trajectories for the trials with added mass show slight irregularities in their motion. Although this non-period-1 motion may be due to unmodeled dynamics, observations made in the vertical hopper analysis suggest that this behavior may be due to the instability described in Section 3.4.1, which occurs when thrust activation occurs near maximum compression and deactivation occurs near take-off.

Finally, the right-most column of experimental trials shows "Hop-settle-fire" hopping trajectories, which were observed in simulation due to the inclusion of damping and gravity. These trajectories result when the stride period is increased such that the hopper starts to settle before thrust is initiated and are sub-optimal in terms of hopping height, as shown in the plots.

### **3.6 Chapter Conclusions**

This chapter has analyzed a one-degree-of-freedom model of a vertical hopper controlled by an open-loop clock-driven force motor pattern. Analytical expressions of the Poincare Maps and associated Jacobians of the model were derived for hopping trajectories that correspond to the two sequences of modes of interest: "Long Thrust" and "Short Thrust" hopping. The steady-state condition was then applied to the Poincare Map, and the performance and stability of the resulting continuum of solutions were analyzed. The following conclusions were drawn:

- Performance for a vertical hopper is correlated to the net amount of work performed by the actuator. Depending on the relative phasing of the activation pattern to the motion of the system in a given steady-state trajectory, the actuator may perform positive and negative work. Trajectories in which maximum work is done correspond to trajectories of maximum hopping height.
- The work performed by the actuator can be further characterized by observing the state at thrust activation and deactivation. If the thrust duration is long enough such that deactivation can occur after take-off, work is maximized by activating thrust at

maximum compression. If the thrust duration is limited, work is maximized by placing thrust deactivation near take-off, which may place thrust activation after maximum compression.

- For open-loop control, multiple steady-state trajectories may exist for a given period of the motor pattern, a result not previously reported. This results in a loss of control, as the system may converge to any one of the solutions available. Results indicate that multiple solutions exist near solutions of optimal hopping height, and that the system will often converge to sub-optimal, but more stable, trajectories.
- Unstable steady-state trajectories can be characterized by the changes in the work performed by the actuator due to perturbations in the timing of thrust activation and deactivation. For the "Long Thrust" case, analytical and numerical results show that trajectories in which thrust is initiated shortly after maximum compression are unstable, as changes in the thrust timing result in unfavorable changes in the work performed that increase the timing perturbation in the subsequent stride cycle. For the "Short Thrust" case, analytical and numerical results show that trajectories with short periods (or high-frequencies) can be unstable with an eigenvalue less than -1. This large negative eigenvalue indicates that perturbations in the thrust timing cause changes in the work performed that over-compensate for the perturbation in the next stride cycle. This helps explain the circumstances under which period-2 behavior occurs.
- All of these observations point to a possibly fundamental trade-off between performance and stability in open-loop vertical hopping. As shown, optimal hopping trajectories may be unstable, or they may be unreachable, with an open-loop motor pattern. As a result, the system may need to be operated at sub-optimal stride periods to avoid unstable or undesirable behavior.

Finally, these characterizations were verified in experimental data of a physical, pneumatically-actuated, vertical hopper called the "Dashpod." Experimental hopping trajectories exhibit many of the observed behaviors predicted by the analytical and numerical analysis, despite the more complex non-linearities of the physical platform.

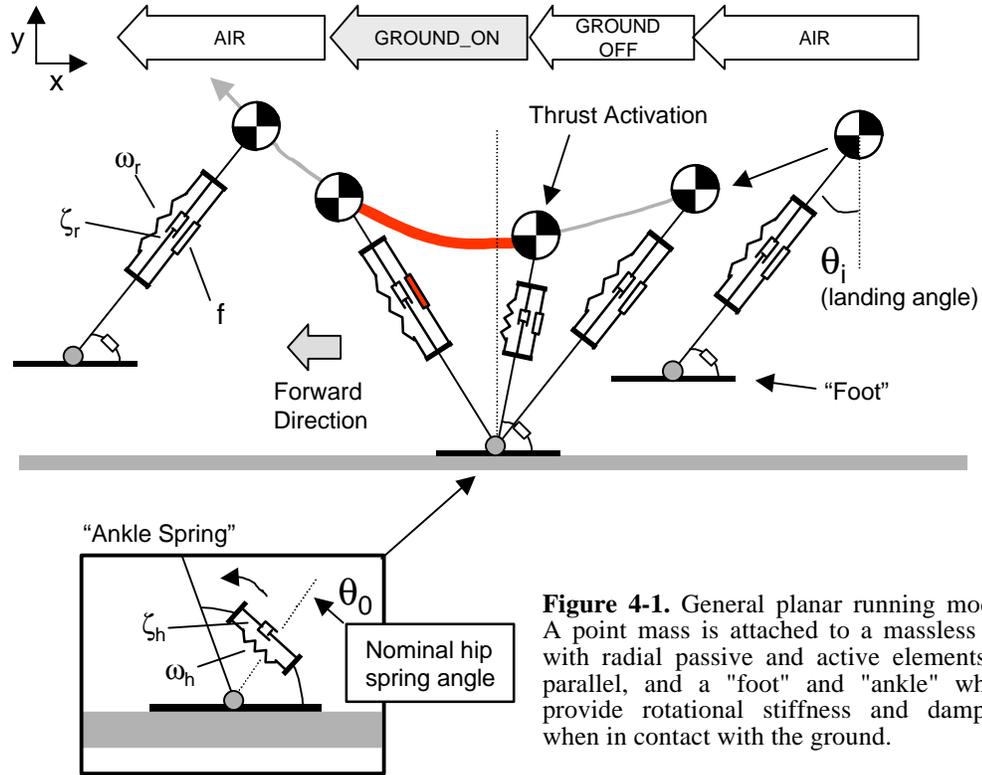
Although the horizontal dynamics of the Sprawl robots play a significant role in their running, the analysis in this chapter provides insight into basic relationships among system energy, actuator timing and stability. These insights are applicable to hopping systems in general, and, as shown in subsequent chapters, play a large role in helping characterize the overall behavior of the Sprawl hexapedal robots.

# 4 Open Loop Planar Running

The basic relationships in the dynamics of open-loop vertical hopping previously established are extended in this chapter through analysis of simplified models of running in the vertical, or sagittal, plane. In particular, we seek direct insights that can help develop strategies for the design and the online "tuning" or adaptation of the Sprawl robots. Since one of the primary goals of locomotion is forward motion, we are interested in performance in terms of forward velocity, and its trade-offs with stability and efficiency. This chapter considers two variations of a simplified planar model. The first, a basic Spring-loaded Inverted Pendulum, or "basic SLIP," is more general, and is intended to add to previous literature in the study of simplified hoppers with novel insights into the factors that determine speed and stability. The second model is intended to capture the basic dynamics of locomotion in the Sprawl robots, and differs from the basic SLIP by an "ankle" spring, which stabilizes motion about the foot, or point of contact with the ground. The first section presents the general model and describes the two model variations. For each of the variations, this chapter first establishes the manner in which performance varies with the timing of the actuator and with system configuration, and then analyses stability based on open-loop control.

## 4.1 Model Description

The general model studied in this chapter is illustrated in Figure 4-1. As shown, the model consists of a point mass that is free to move in the plane and that is attached to a massless telescoping leg with stiffness and damping such that the system has a natural frequency  $\omega_r$ , and damping ratio  $\zeta_r$ , in the radial direction. The spring and damper are in parallel with a force-producing element,  $f(t)$ , which applies a force according to the activation motor pattern. The spring has a nominal rest length of  $r_0$ . In this idealized model, the massless leg is automatically reset to a landing angle,  $\theta_i$ , with respect to the horizontal while the point mass is in the air. The leg comes into contact with the ground when,



**Figure 4-1.** General planar running model. A point mass is attached to a massless leg with radial passive and active elements in parallel, and a "foot" and "ankle" which provide rotational stiffness and damping when in contact with the ground.

$$y < r_0 \cos(\theta_i) \quad (4.1)$$

where  $y$  is the height of the mass measured relative to the floor. The location where the leg contacts the ground determines the location of the leg's foot for the rest of the stance phase. The hopper takes off when the distance from the mass to the foot exceeds  $r_0$ . When in contact with the ground, the leg has an "ankle" with normalized rotational stiffness  $\omega_h$  and damping ratio  $\zeta_h$  that exerts a torque on the leg-mass system that pulls the direction of the leg toward a nominal leg angle, as shown in the figure.

The model's hybrid modes and transition events are as generally described in Equations 2.12 to 2.19. For simplicity, we will only consider the sequence of modes termed "Long Thrust," and assume that the relationships among work, performance and stability for "Short Thrust" trajectories extend in a similar way to the planar case. "Long Thrust" trajectories are those in which thrust by the leg's force-producing element is initiated at some point in the stance trajectory and lasts until take-off:

$$\text{"Long Thrust": } \{ \text{AIR\_OFF, GROUND\_OFF, GROUND\_ON, AIR\_ON} \} \quad (4.2)$$

When the hopper is controlled completely open-loop, all of the system parameters remain constant from one stride to the next, including the landing angle and the nominal radial and hip spring positions, and thrust activation is initiated by the fixed motor pattern.

#### 4.1.1 Equations of Motion

Unlike the vertical hopper, the equations of motion in this model during the stance phase are nonlinear. For numerical simulation, we use the mass' X-Y coordinates, normalized by gravity, and their velocities as the state. Thus, the accelerations are simply given by the sum of forces, normalized by mass and gravity, due to the leg and ankle's passive elements, the thrust force and gravity:

$$\Sigma F = F_{kr} + F_{br} + F_{kh} + F_{kh} + F_f + F_g \quad (4.3)$$

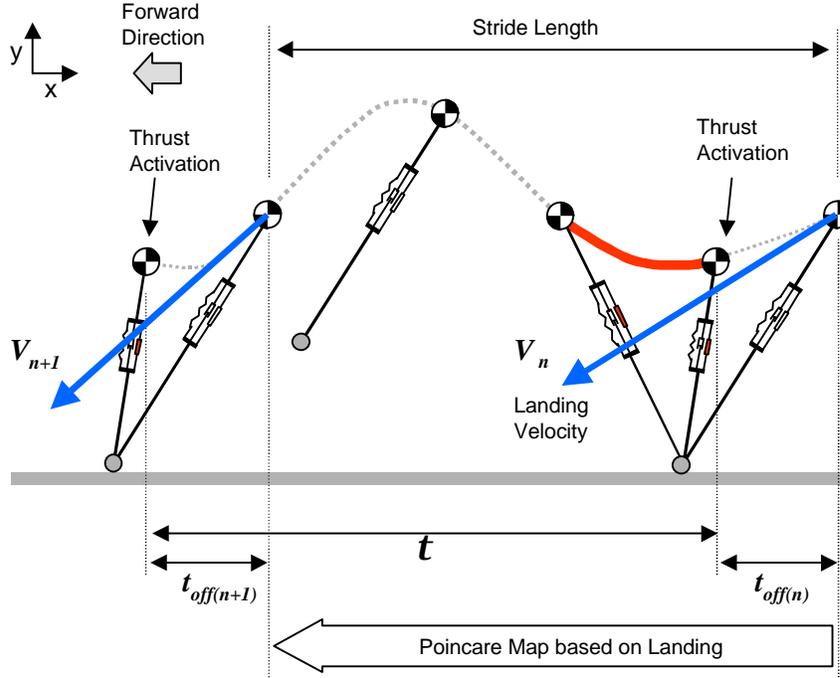
Each individual normalized force is defined by:

$$\begin{aligned} \text{Force due to radial spring: } F_{kr} &= -u_r \cdot \omega_r^2 \cdot (r - r_0) \\ \text{Force due to radial damper: } F_{br} &= -u_r \cdot 2\zeta_r \omega_r \cdot \left( u_r \bullet \begin{bmatrix} v_x & v_y \end{bmatrix}^T \right) \\ \text{Force due to ankle spring: } F_{kh} &= -u_t \cdot \omega_h^2 \cdot (\theta - \theta_0) / r \\ \text{Force due to radial damper: } F_{bh} &= -u_t \cdot 2\zeta_h \omega_h \cdot \left( u_t \bullet \begin{bmatrix} v_x & v_y \end{bmatrix}^T \right) / r^2 \\ \text{Force due to radial thrust: } F_f &= u_r \cdot f(t) \\ \text{Force due to gravity: } F_g &= \begin{bmatrix} 0 & -1 \end{bmatrix}^T \end{aligned} \quad (4.4)$$

where  $u_r$  is the unit vector along the leg,  $u_t$  is the unit tangent vector that is normal to  $u_r$ ,  $v_x$  and  $v_y$  are the horizontal and vertical components of the velocity of the mass,  $r$  is the instantaneous leg length, given by the distance of the mass to the foot, and  $\theta$  is the angle of the leg with respect to the vertical.  $f(t)$  is the thrust force normalized by mass and gravity, and is given by:

$$f(t) = \begin{cases} f & t_{off} < t \leq t_{off} + t_{on} \\ 0 & otherwise \end{cases} \quad (4.5)$$

where  $t_{off}$  is the time between landing and thrust activation and  $t_{on}$  is the thrust duration.



**Figure 4-2.** Poincaré Map between landing events. The reduced state vector mapped consists of the velocity of the mass (magnitude and direction), and  $t_{off}$ , the time between landing and thrust activation.

### 4.1.2 Poincaré Map and Jacobian

For this system, a possible Poincaré Map is one that maps the state at thrust activation from one stride period to the next, relative to the location of the foot. However, this map would have to account for the fact that not all states, X-Y position and velocities, are possible given an initial state at landing with the leg as prescribed by the landing angle,  $\theta_i$ . Additional constraint equations would have to accompany the Poincaré Map. Thus, it becomes more convenient to use a Poincaré Map based on the state at landing, where the reduced state vector that is mapped consists of the velocity at landing (magnitude,  $|V|$ , and direction  $\theta_V$ ), and  $t_{off}$ , the time between landing and thrust activation, as shown in Figure 4-2, and as described by:

$$\tilde{X}_{L(n+1)} = M(\tilde{X}_{L(n)}) \quad (4.6)$$

$$\tilde{X}_L = \begin{bmatrix} \theta_V \\ |V| \\ t_{off} \end{bmatrix} \quad (4.7)$$

As shown later in this chapter, choosing these particular polar coordinates for the landing velocity reveals certain properties of the stability of the system. Since the equations of motion cannot be integrated analytically, the function  $M$  here is a dynamic numerical simulation of the model. The simulation starts at the initial condition given by the landing angle and the state vector  $\tilde{X}_L$ , and terminates the next instant that landing occurs. Fixed points are found by solving numerically for the state vector that satisfies:

$$\tilde{X}_L^* - M(\tilde{X}_L^*) = 0 \quad (4.8)$$

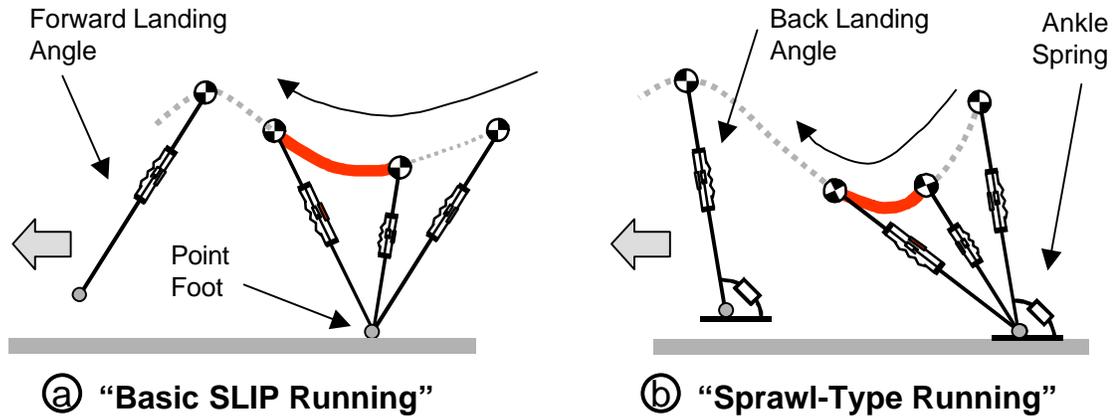
The Jacobian is also found numerically through simulation. Each column vector of the Jacobian is found by first introducing a disturbance about the fixed point in one of the state vector's elements, and simulating until the next landing event. Deviations from the fixed point at this landing are measured, and divided by the original disturbance, resulting in an approximation of the partial derivative with respect to that element of the state vector. For example, the third column vector of the Jacobian is approximated as:

$$\frac{\partial \tilde{X}_{L(n+1)}}{\partial t_{off(n)}} \approx \frac{\Delta \tilde{X}_{L(n+1)}}{\Delta t_{off(n)}} \approx \frac{M\left(\tilde{X}_L^* + \begin{bmatrix} 0 & 0 & \Delta t_{off(n)} \end{bmatrix}^T\right) - M(\tilde{X}_L^*)}{\Delta t_{off(n)}} \quad (4.9)$$

and the Jacobian is composed as:

$$\frac{\partial \tilde{X}_{L(n+1)}}{\partial \tilde{X}_{L(n)}} \approx \begin{bmatrix} \frac{\Delta \tilde{X}_{L(n+1)}}{\Delta \theta_{V(n)}} & \frac{\Delta \tilde{X}_{L(n+1)}}{\Delta |V|_{(n)}} & \frac{\Delta \tilde{X}_{L(n+1)}}{\Delta t_{off(n)}} \end{bmatrix} \quad (4.10)$$

In the subsequent analyses, for each column of the Jacobian, a range of disturbances varying in sign and magnitude were used, and the resulting deviations were averaged to obtain the final values.



**Figure 4-3.** Illustration of "Basic SLIP" and "Sprawl-type" variations on the planar running model. The Basic SLIP has no ankle compliant element, and is characterized by trajectories in which the leg is placed forward for landing. In the "Sprawl-type" running model, a compliant ankle stabilizes the direction of the leg, and the leg is placed back for landing.

## 4.2 Basic SLIP Running

The first variation studied here of the general model presented above is the "basic SLIP" or basic Spring-loaded Inverted Pendulum. For this model, we ignore the ankle spring, assuming that the leg's foot acts essentially as a pin joint when in contact with the ground, as illustrated in Figure 4-3a. Without the ankle spring, the stance dynamics are essentially those of an unstable inverted pendulum. SLIP models have been at the heart of the study of dynamic locomotion from early in its history.

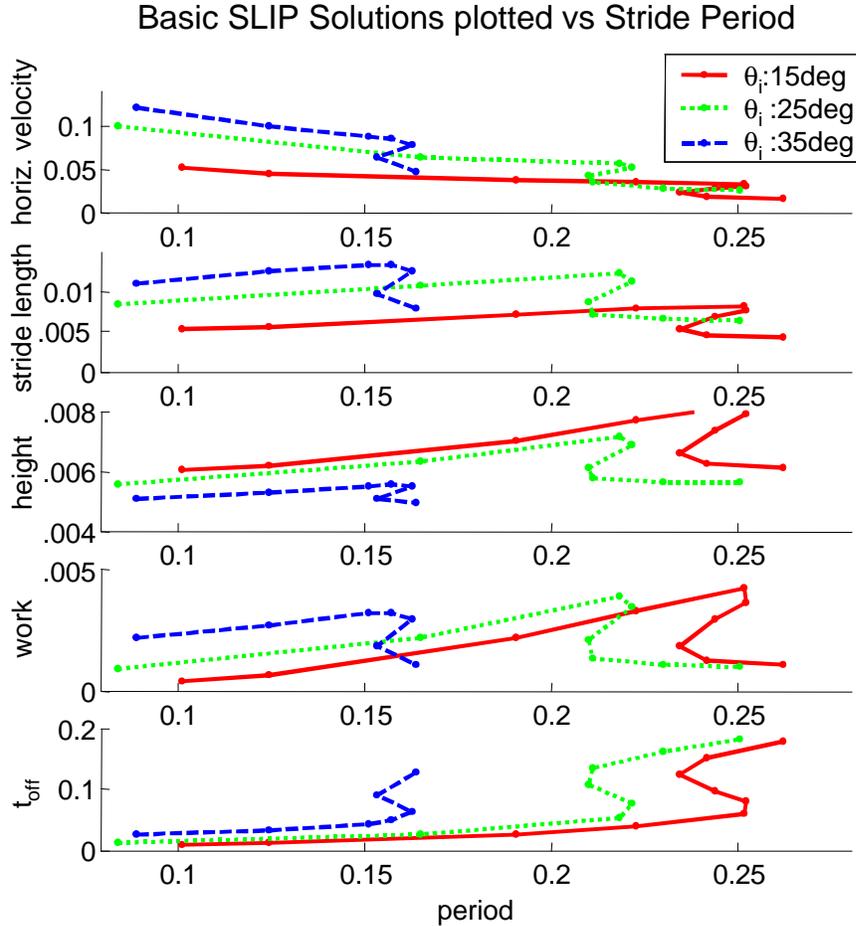
Raibert's monopod hoppers (Raibert, 1986) were essentially SLIPs, and they were capable of stable forward motion using a control framework that decoupled vertical motion from horizontal motion. Vertical motion was assumed to be similar to the vertical hoppers studied in the previous chapter, and controlled closed-loop by initiating thrust when the leg was at maximum compression. Steady-state hopping height was controlled by adjusting thrust duration. Control of motion in the horizontal direction was based on the concept of "neutral foot positions," which held that for a given horizontal landing velocity, there exists a leg landing angle such that horizontal velocity was neither increased or decreased during stance. Forward motion was then controlled in closed-loop by placing the leg before the neutral position to accelerate, and after the neutral position to decelerate. This simple control law worked remarkably well to stabilize the system in running.

The SLIP model is also the basis for the hypothesis that animals of different size, morphologies, leg number and leg arrangement all move dynamically like a SLIP (Full and Koditschek, 2000), in terms of the ground reaction forces measured when running. Dynamic measurements of Sprawlita, one of the Sprawl robots, using motion capture and a force measuring platform (Cham et al., 2001) show that the ground reaction forces are similar to those of a basic SLIP.

Previous analyses of the SLIP model have focused almost exclusively on versions of the SLIP controlled using Raibert's decoupled, closed-loop control laws (see for example, McCloskey and Burdick, 1993). The open-loop monopod of Mombaur (2000) was used mainly to show that open-loop stable trajectories exist, and did not explore the effects of changing the open-loop motor pattern. Therefore, given the significance of the SLIP in our understanding of dynamic locomotion, an important question that has not been addressed is whether there exist other ways that thrust activation and leg landing angle in a SLIP can be controlled. If the SLIP cannot be controlled open-loop, are there alternative closed-loop approaches to stabilizing the system? The following sections explore these questions.

### **4.2.1 Performance in Basic SLIP Running**

The idea that there may be advantages to initiating thrust in a hopper at other points in the cycle besides maximum compression has been largely ignored. An important result of the previous chapter was that, for a "Short Thrust" vertical hopper, initiating thrust at maximum compression is suboptimal in terms of hopping height. In analyzing the SLIP "Long Thrust" model, fixed point solutions for a range of landing angles,  $\theta_i$  were searched for. For each landing angle, the variable  $t_{off}$  at landing was varied, and the landing velocity magnitude and direction that resulted in the same velocity magnitude and direction upon the next landing were found by numerically integrating the equations of motion using the "ode23" function in Matlab<sup>TM</sup> with a tolerance of  $10^{-6}$ . For each solution, the stride length (horizontal distance covered within one stride), the period (time between landings), the hop height (maximum height in airborne phase), and the work performed by the actuator within that stride were measured. Varying the value of  $t_{off}$  allows us to find solutions other than the ones in which thrust is initiated at maximum compression. Values for the leg spring,

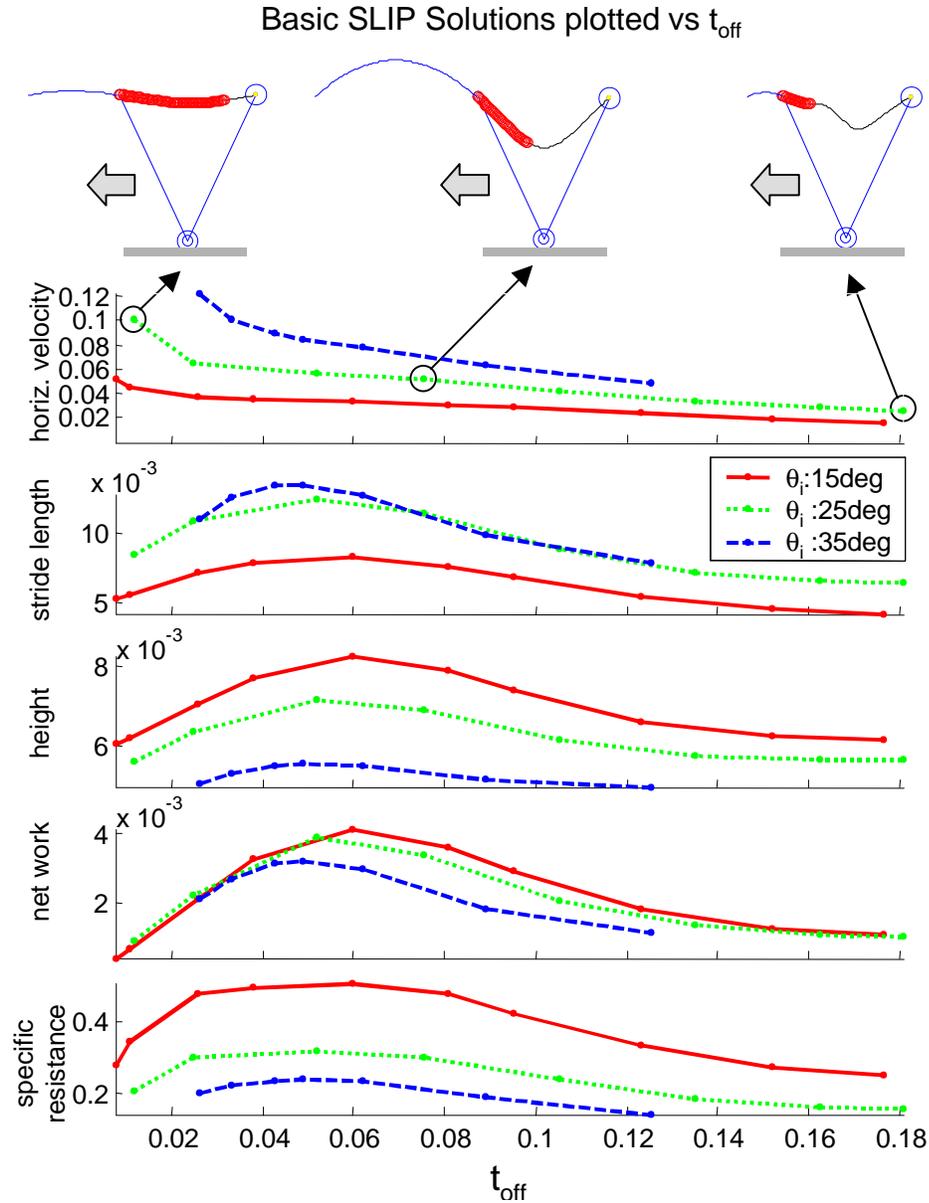


**Figure 4-4.** Steady-state solutions for Basic SLIP model with varying leg landing angle. Note that for open-loop control, there exist multiple solutions for a range of stride periods.

damper and force element were  $\omega = 30$ ,  $\zeta = 0.2$  and  $f = 1.5$ . The search results are illustrated in Figure 4-4 plotted against the period, and in Figure 4-5 plotted against  $t_{off}$ .

As shown in Figure 4-4, there exists a continuum of solutions for each landing angle as the moment of thrust initiation is varied. In addition, multiple solutions exist when the continuum is plotted against the stride period. Similar to the vertical hopper case, this is attributed to the fact that solutions with long airborne phases achieve a local maximum in stride period, such that nearby solutions will share the same periods. This shows that, as in the vertical hopper, the use of open-loop control results in the loss of control over the resulting behavior, as the system may converge to any one of the available solutions.

Multiple solutions do not appear in the continuum when plotted against  $t_{off}$ , again indicating that a thrust activation controller based on sensing landing and using  $t_{off}$  would



**Figure 4-5.** Steady-state solutions for Basic SLIP plotted against  $t_{off}$ . Solutions in the mid-ranges of  $t_{off}$  represent trajectories in which thrust is initiated near maximum compression and have longer stride lengths and hopping height, but are suboptimal in terms of forward speed. Solutions in which thrust is initiated shortly after landing, with minimal  $t_{off}$ , have higher forward velocities.

avoid multiple solutions, requiring only a simple contact switch. This possibility is addressed in a subsequent section.

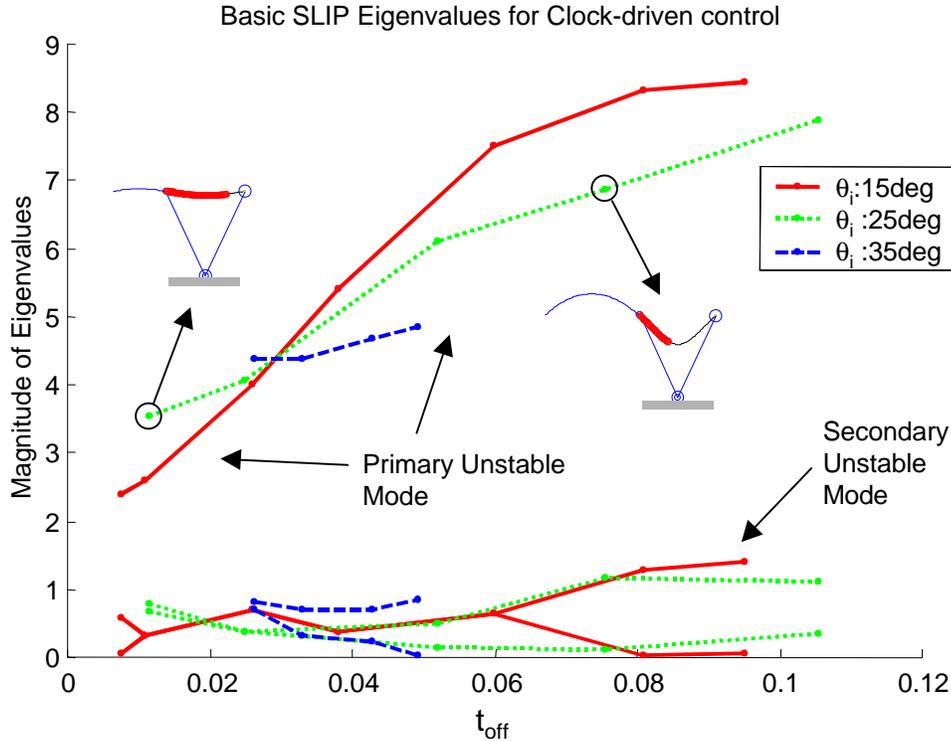
As shown by the sample hopper trajectories in Figure 4-5, the continuum spans solutions in which thrust is initiated shortly after landing to solutions in which thrust is initiated well into the stance phase, such that motion in the radial direction has started to settle.

The figure shows that forward velocity is a monotonically decreasing function of  $t_{off}$ , such that it is maximized for low values of  $t_{off}$ . This indicates that solutions in which thrust is initiated shortly after landing, here termed "skittering" solutions, are more optimal in terms of horizontal velocity for a given leg landing angle than solutions in which thrust is initiated at maximum compression. By initiating thrust early, these "skittering" solutions are able to sustain a large horizontal landing velocity in steady-state by supporting the mass without significant leg compression, in a motion that is similar to "vaulting" with a stiff pole. The figure also shows that these solutions do not have the highest stride length (distance covered per stride), but due to the shorter stride period, and thus higher stride frequency, the solutions have higher horizontal speeds.

Stride length is maximized in solutions in which thrust is initiated near maximum compression of the leg. As in the vertical hopper case, this is attributed to a corresponding maximum amount of net work performed by the actuator. This maximum in stride length also corresponds to a maximum in hopping height. Solutions with significant leg compression appear to result in take-off angles closer to a favorable angle of 45 degrees, such that both the length and height of the airborne phase are maximized.

Specific resistance (Gabrielli and von Karman, 1950), a measure of the efficiency of locomotion, is given by the ratio of the energy expended in a stride to the product of mass, gravity and the stride length. As shown in the figure, specific resistance is nearly constant for the range of solutions in which thrust is initiated near maximum compression, as activating slightly after or before maximum compression results in both shorter stride length and less energy expended. As thrust is activated further away from maximum compression, specific resistance decreases. Although stride lengths are shorter for these trajectories, the vertical component of the landing velocity is smaller such that significantly less work is used to reverse the vertical motion of the mass during stance.

Leg landing angle has a significant effect on speed, as shown also in Figure 4-5. Increasing the landing angle increases speed and stride length, and decreases the hopping height. From Figure 4-4, note that changing the leg angle significantly changes the period for which stride length is maximized. However, this period maintains correspondence with the period for which work is maximized. Leg landing angles of 40 degrees and more did



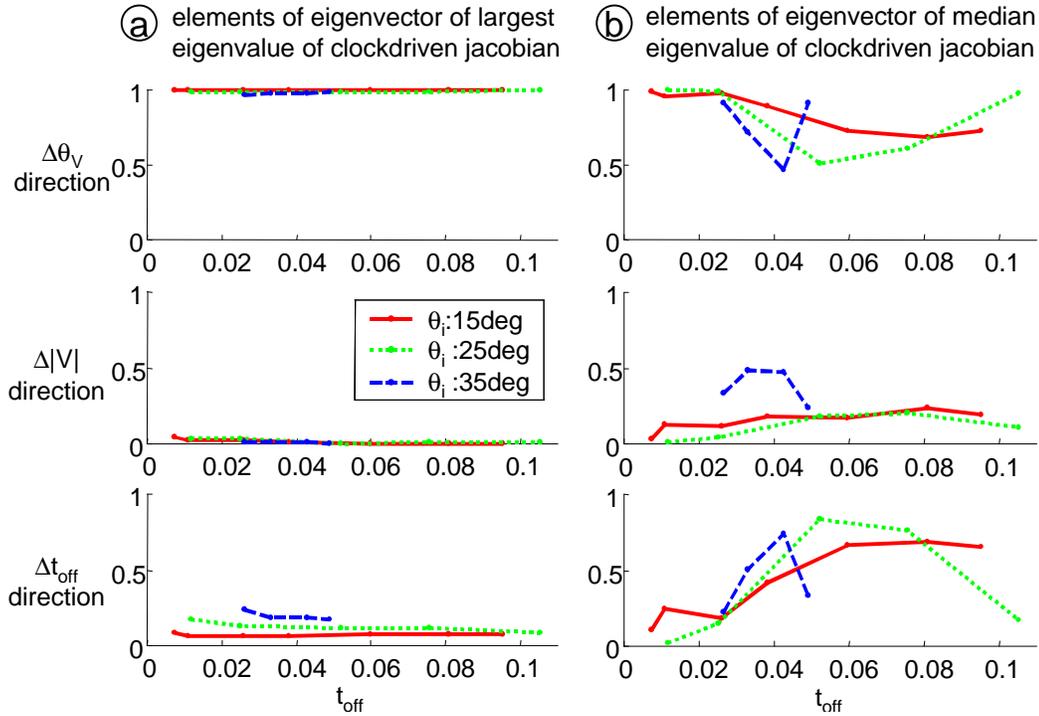
**Figure 4-6.** Eigenvalues of the Jacobian for the case of open-loop control of the SLIP. As shown, "skittering" hopping trajectories are less unstable, and instability is dominated by one mode.

not result in any solutions, indicating there while speed is increased with leg angle, there is a limit for which steady-state solutions will exist.

These observations relate to steady-state trajectories of the basic SLIP model, and are thus relevant to hoppers in general, regardless of whether thrust and the leg landing angle are controlled close-loop or open-loop. The following section studies the stability of these trajectories under open-loop control.

#### 4.2.2 Stability of Basic SLIP

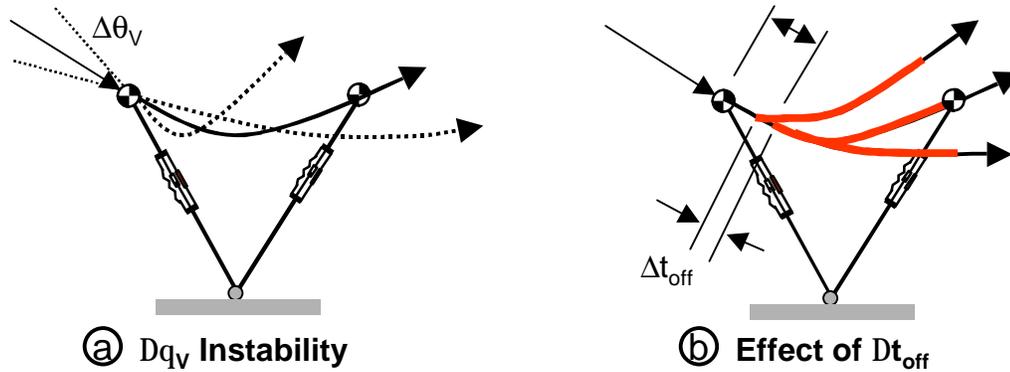
To evaluate whether the Basic SLIP could be stably operated with open-loop control, we look at the Jacobian for the case of a fixed motor pattern, which is found through simulation as given by Equation 4.10. Again, open-loop control means that the leg landing angle is reset to the same position during each airborne phase, and that thrust activation is given by the fixed motor pattern. The eigenvalues of the Jacobian for this case of the SLIP are shown in Figure 4-6 for the three landing angles, plotted against  $t_{off}$ .



**Figure 4-7.** Magnitudes of the elements of the eigenvector corresponding to the largest and median eigenvalues. As shown, the unstable mode with the largest eigenvalue is primarily in the direction of disturbances in the angle of the landing velocity. The unstable mode that appears for longer values of  $t_{off}$  is primarily in the direction of disturbances in  $t_{off}$ .

As shown, all solutions are unstable. Instability is in large part due to one primary unstable mode, given by the eigenvalue with large magnitudes that range from 2 to 9 shown in the figure. This eigenvalue can be shown to correspond to a mode that is in the direction of disturbances in  $\theta_V$ . Figure 4-7a shows the magnitudes of the elements of the unit eigenvector corresponding to the large, unstable eigenvalue. As shown in the figure, the eigenvector points primarily in the direction of  $\Delta\theta_V$ . Figure 4-8a shows the nature of this unstable mode, wherein for certain steady-state trajectories small disturbances in the angle of the landing velocity result in large deviations from the nominal trajectory. As shown in Figure 4-6, the magnitude of this unstable mode decreases for "skittering" trajectories with small  $t_{off}$ .

From Figure 4-6, we see that another secondary unstable mode arises for longer values of  $t_{off}$ . This instability can be shown to correspond to the same instability reported for the "Long Thrust" vertical hopper in the previous chapter, which is related to the timing of thrust activation relative to landing. Figure 4-7b shows the elements of the eigenvector



**Figure 4-8.** a) Unstable mode in the direction of the landing velocity for the Basic SLIP. Disturbances in the direction of the landing velocity result in large perturbations in the direction of the take-off velocity. b) Effect of disturbances in  $t_{off}$ , the time of thrust activation. It can be shown that  $t_{off}$  can be used in a control law that stabilizes a hopper with open-loop leg landing angle.

that corresponds to the median eigenvalue. As shown, for  $t_{off}$  values near the instability, the third element of the eigenvector,  $\Delta t_{off}$ , increases. These values of  $t_{off}$  correspond to the solutions in which thrust is initiated shortly after maximum compression of the leg.

As shown, operating in "skittering" trajectories shows promise that stability could be achieved open-loop. Both the instability due to  $\Delta\theta_V$  and the instability due to timing are reduced for trajectories in which thrust is initiated shortly after landing. Determining the existence of an open-loop stable SLIP solution is left for future work.

### 4.2.3 Alternate Stabilization of the Basic SLIP

The above analysis found that by varying the moment in the cycle that thrust is activated, a range of trajectories of the Basic SLIP model not previously considered is revealed, and that hopping behavior can vary a great deal within this range. For example, the same horizontal velocity achieved by using a steep leg landing angle and thrusting at maximum compression can also be achieved by using a more vertical landing angle and thrusting before maximum compression. The first configuration could be useful in situations where high ground clearance is needed, as the resulting trajectories also maximize hopping height. The second configuration could be useful in situations where ground traction is critical, as steep landing angles could result in slippage.

Unfortunately, none of these trajectories were found to be stable under open-loop control. However, the discovery of this previously ignored "knob" that varies the instant

that thrust occurs leads one to ask whether it can be used to stabilize the basic SLIP. As shown in the previous section, the basic SLIP is unstable in large part due to the instability in the angle of the landing velocity that results when the leg landing angle is fixed from stride to stride. Can a basic SLIP with fixed landing angle be stabilized by controlling the timing of thrust activation? Although the answer is not directly consequential to our understanding of open-loop locomotion, we take an aside to show that such stabilization is possible, thereby providing an alternative to Raibert's method for the control of the basic SLIP.

We first assume a hopper in which the leg landing angle is fixed, but thrust activation is controlled by a clock timer that is reset to zero at landing. When the timer reads a value equal to  $t_{off}$ , thrust is activated, and remains active until take-off. The value of  $t_{off}$  can be set by the controller. Changing  $t_{off}$  from its steady-state value has a significant effect on the trajectory of the system, as illustrated in Figure 4-8b. Formulating a Poincare Map for this controller about the instant of landing, we can write the following discrete system for disturbances about a steady-state trajectory:

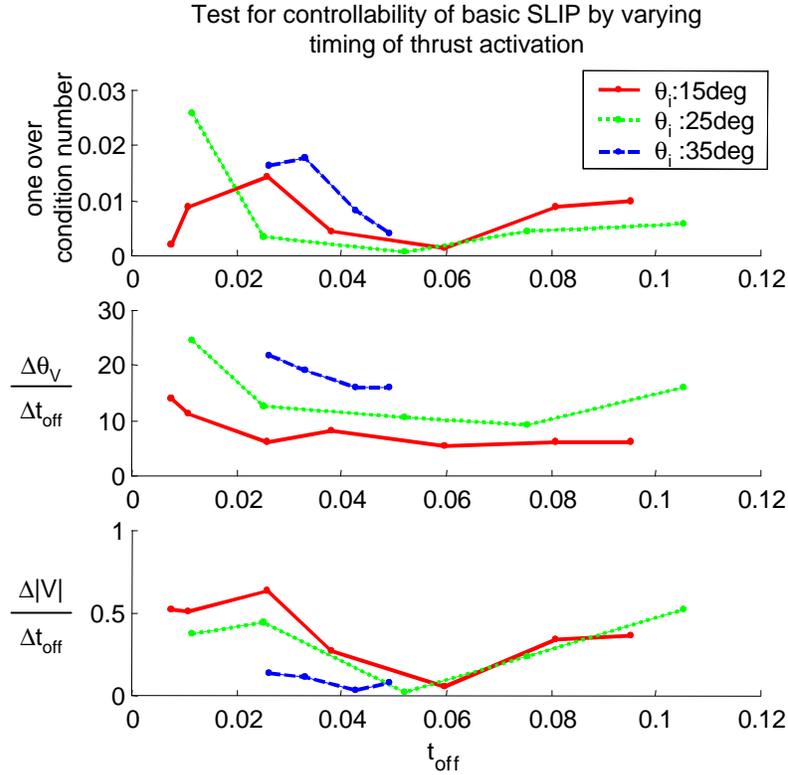
$$\begin{bmatrix} \Delta\theta_{V(n+1)} \\ \Delta|V|_{(n+1)} \end{bmatrix} = \begin{bmatrix} \frac{\partial\theta_{V(n+1)}}{\partial\theta_{V(n)}} & \frac{\partial\theta_{V(n+1)}}{\partial|V|_{(n)}} \\ \frac{\partial|V|_{(n+1)}}{\partial\theta_{V(n)}} & \frac{\partial|V|_{(n+1)}}{\partial|V|_{(n)}} \end{bmatrix} \begin{bmatrix} \Delta\theta_{V(n)} \\ \Delta|V|_{(n)} \end{bmatrix} + \begin{bmatrix} \frac{\partial\theta_{V(n+1)}}{\partial t_{off(n)}} \\ \frac{\partial|V|_{(n+1)}}{\partial t_{off(n)}} \end{bmatrix} \Delta t_{off(n)} \quad (4.11)$$

Since  $t_{off}$  is controlled,  $\Delta t_{off(n)}$  is not included in the state of the Poincare Map, and is instead treated as a control input. This system has the form:

$$X_{n+1} = PX_n + Tu \quad (4.12)$$

Showing that this linearized discrete system is controllable implies that the basic SLIP can be stabilized by the control input  $\Delta t_{off(n)}$ . Controllability, as given by Franklin, et al. (1997), of a discrete system of the form of Equation 4.11 implies there exists a control law,

$$u = KX_n \quad (4.13)$$

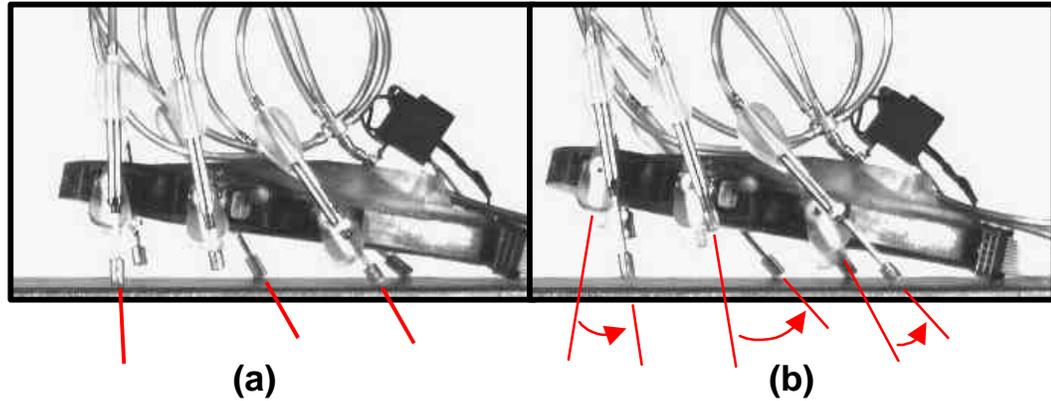


**Figure 4-9.** Inverse of the condition number for the controllability matrix. Non-zero values indicate that the system is controllable, and that the basic SLIP can be stabilized by controlling the thrust timing  $t_{off}$ . As shown, for mid-range values of  $t_{off}$  which correspond to initiating thrust near maximum compression, the effect of  $t_{off}$  on the magnitude and direction of the landing velocity decreases, and  $t_{off}$  loses "control authority."

that can place the poles of the system in any desired location. Controllability in this case can be established by showing that the following matrix,  $C$ , is full rank:

$$C = \begin{bmatrix} T & PT \end{bmatrix} \quad (4.14)$$

The matrices  $T$  and  $P$  were found using the appropriate elements of the Jacobian, given by Equation 4.10, which were already found by simulation. Figure 4-9 shows results of testing the rank of  $C$  for the continuum of solutions previously presented. The top plot show the inverse of the condition number of the matrix  $C$  for each solution. Values near zero for this inverse indicate that the matrix is nearly singular, and not full rank. As shown, this inverse is non-zero for low values of  $t_{off}$  but nearly zero for mid values. Although these numerical results are subject to noise and small inaccuracies due to the linearization, they generally imply that the system is controllable for a range of solutions, and that the basic

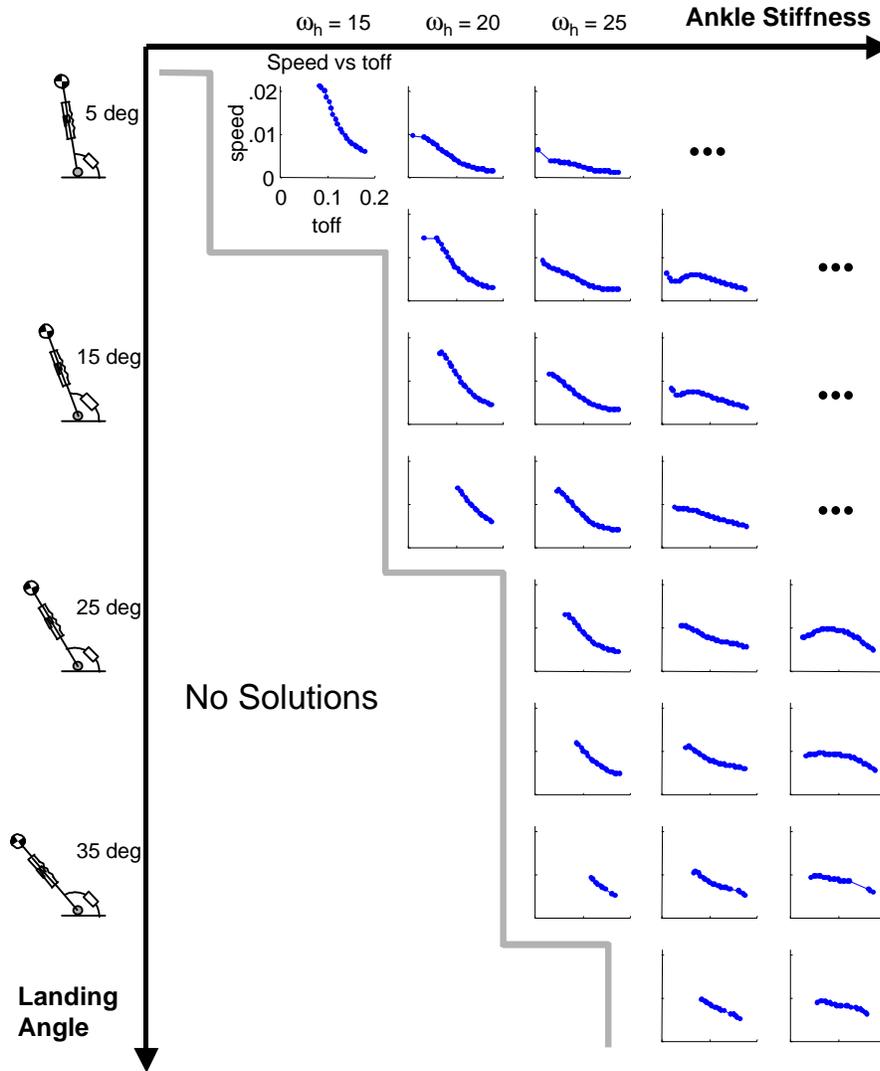


**Figure 4-10.** Captured images of Sprawlita in locomotion. As shown, the hip flexures play a large role in locomotion, and undergo significant deflection during stance.

SLIP can be stabilized by  $t_{off}$ . Solutions for which the system is not controllable can be explained by plots of  $\Delta\theta_{V(n+1)}/\Delta t_{off(n)}$  and  $\Delta|V|_{(n+1)}/\Delta t_{off(n)}$ , which decrease significantly for this mid range. In this range, thrust activation occurs near maximum compression, where velocities are low, and changes in  $t_{off}$  have a lesser effect in the trajectory. The system simply loses "control authority." Nevertheless, this examination shows that, at least for a range of solutions, it is possible to stabilize a basic SLIP with an open-loop leg landing angle, using only changes in  $t_{off}$  as the control input, based on feedback from errors in the landing velocity.

### 4.3 Sprawl-type Running

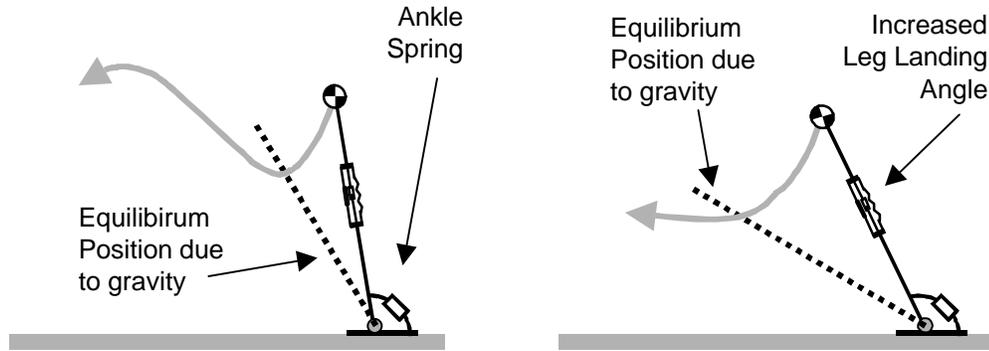
Although dynamic measurements of Sprawlita, one of the Sprawl robots, shows similarities in the ground reaction forces and motion of the center of mass to the basic SLIP (Cham et al., 2001), the specific mechanism for locomotion in the Sprawl robots may not be best captured by the simple SLIP. In the Sprawl Robots, the compliant elements at the robot's hips play a large role in the locomotion, and the telescoping legs act largely as struts or thrusters. High speed video of the Sprawl robots while running helps characterize the motion, as shown in Figure 4-10. During locomotion, the legs contact the ground at an angle set by the nominal angle of the hip springs. This angle is nearly vertical for the front legs, and set back for the middle and hind legs. During stance, the legs extend while thrusting, and the hips undergo significant deflection. At take-off, or when the legs are deactivated, the energy stored in the hip flexures is dissipated as the legs return to their nominal position, ready for



**Figure 4-11.** Forward speed of the steady-state solutions for a range of ankle stiffnesses and leg landing angles. As shown, speed is maximized at the borderline of the region for which solutions exist.

the next contact phase. A variation on the basic model that attempts to capture this mechanism is the focus of the rest of the chapter.

The second model studied in this chapter is illustrated in Figure 4-3b. The model differs from the basic SLIP in the inclusion of a foot with an ankle spring and damper, and in that the leg landing angles are set back in relation to the forward motion. The ankle spring in the simplified model is intended to capture the combined effects of the hip flexures and the sprawled posture of the Sprawl robots. The sprawled posture helps maintains the body level, while the hip springs absorb energy as the legs are rotated about the contact point.



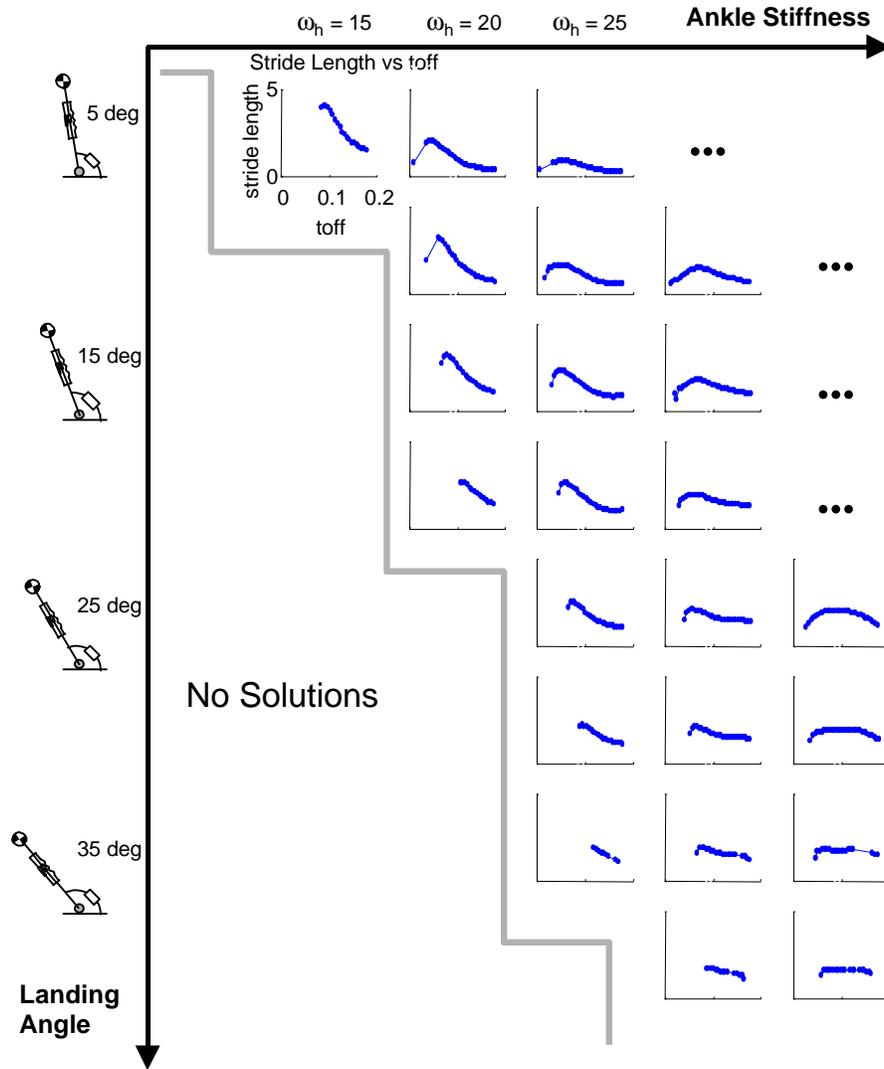
**Figure 4-12.** Effect of increasing the leg landing angle. By increasing the landing angle, the equilibrium position of the ankle increases, due to the increased effect of gravity. If the equilibrium position is too steep, the system cannot recover and no solution will exist. Decreasing the ankle stiffness has a similar effect.

Since the flexures consist of visco-elastic material, the ankle spring includes a damping element. The nominal position of the ankle spring is the same as the landing angle.

For the Sprawl robots, we are motivated to understand the effects of changing the nominal leg angles and the period and duty cycle of the activation pattern on locomotion. We are also interested in finding guidelines for the design of the robot's hip flexures, in terms of stiffness and damping properties. To gain insights into these problems, steady-state solutions were found for the model for a range of ankle stiffnesses, leg landing angles and stride period, using the equations of motion and Poincare Map described in the beginning of the chapter.

### 4.3.1 Performance and Stability

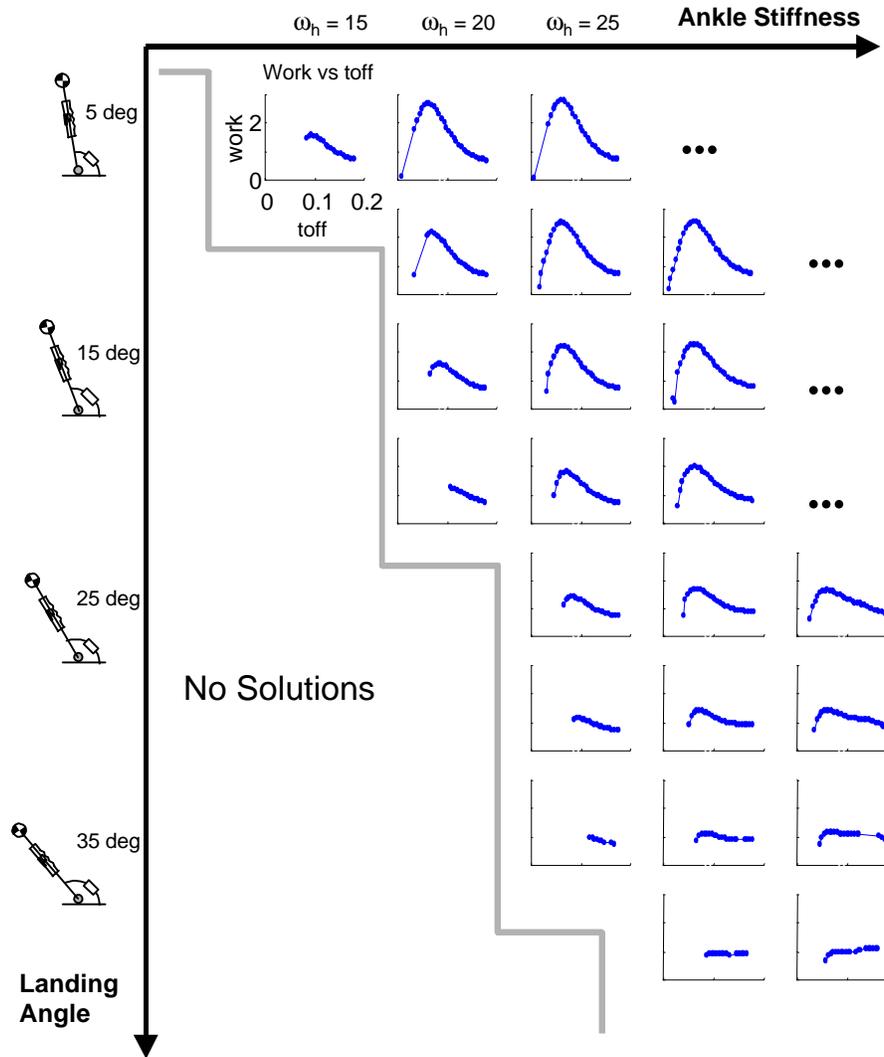
The solutions found are shown in the grid of plots in Figure 4-11. In the figure, each element in the grid represents a combination of ankle stiffness and leg landing angle, and shows the continuum of solutions for that combination in terms of forward velocity plotted against  $t_{off}$ . A first observation is that for certain combinations of landing angles and ankle stiffness there exist no steady-state solutions. This region of no solutions is shown in the figure, and has a boundary that is diagonal across landing angle and ankle stiffness. This means that for a given stiffness, there is a maximum landing angle for which solutions exist, and for a given landing angle, there is a minimum ankle stiffness. This is illustrated in Figure 4-12. In the Sprawl robots, it has been found experimentally that the robots "nose-dive" during locomotion if the hip flexures are too soft or the leg angles are set too far back.



**Figure 4-13.** Stride length of the steady-state solutions for a range of ankle stiffnesses and leg landing angles. As shown, stride length is not maximized at the same values of  $t_{off}$  as forward speed.

The simulation results suggest that this undesirable nose-diving motion is the result of the non-existence of steady-state solutions, and not the result of an existing, locally unstable, solution. This is an important distinction, as unstable trajectories have the possibility of being stabilized under closed-loop control. A lack of solutions precludes the possibility of steady-state motion.

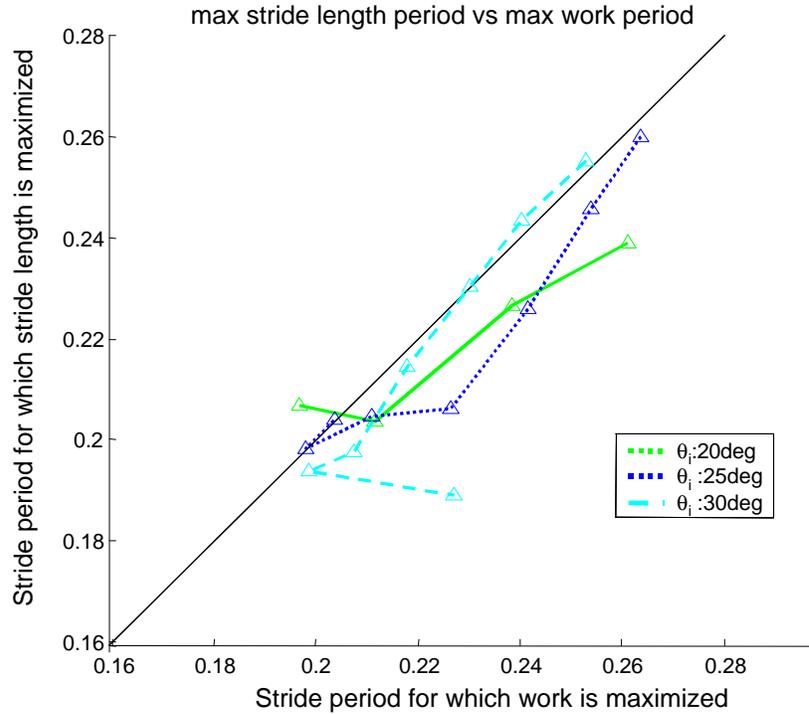
A second observation is that for combinations in which solutions exist, there is a minimum value of  $t_{off}$  below which no solutions exist. Since the leg landing angle is set



**Figure 4-14.** Work performed by the actuator in the steady-state solutions for a range of ankle stiffnesses and leg landing angles. As shown, combinations of ankle stiffness and leg landing angle with maximum peak work do not correspond to combinations with maximum peak stride length. However, for each particular combination and ankle stiffness and landing angle, the optimal value of  $t_{off}$  which maximizes stride length is near the same  $t_{off}$  which maximizes work.

back, this form of hopping relies heavily on the ankle spring to decelerate the vertical component of the velocity at landing. Steady-state solutions require a minimum  $t_{off}$  that allows the ankle spring to contribute to absorbing the kinetic energy before thrust, and subsequent take-off, can occur. Therefore, this minimum  $t_{off}$  increases with softening ankle stiffness, and with increasing leg angle.

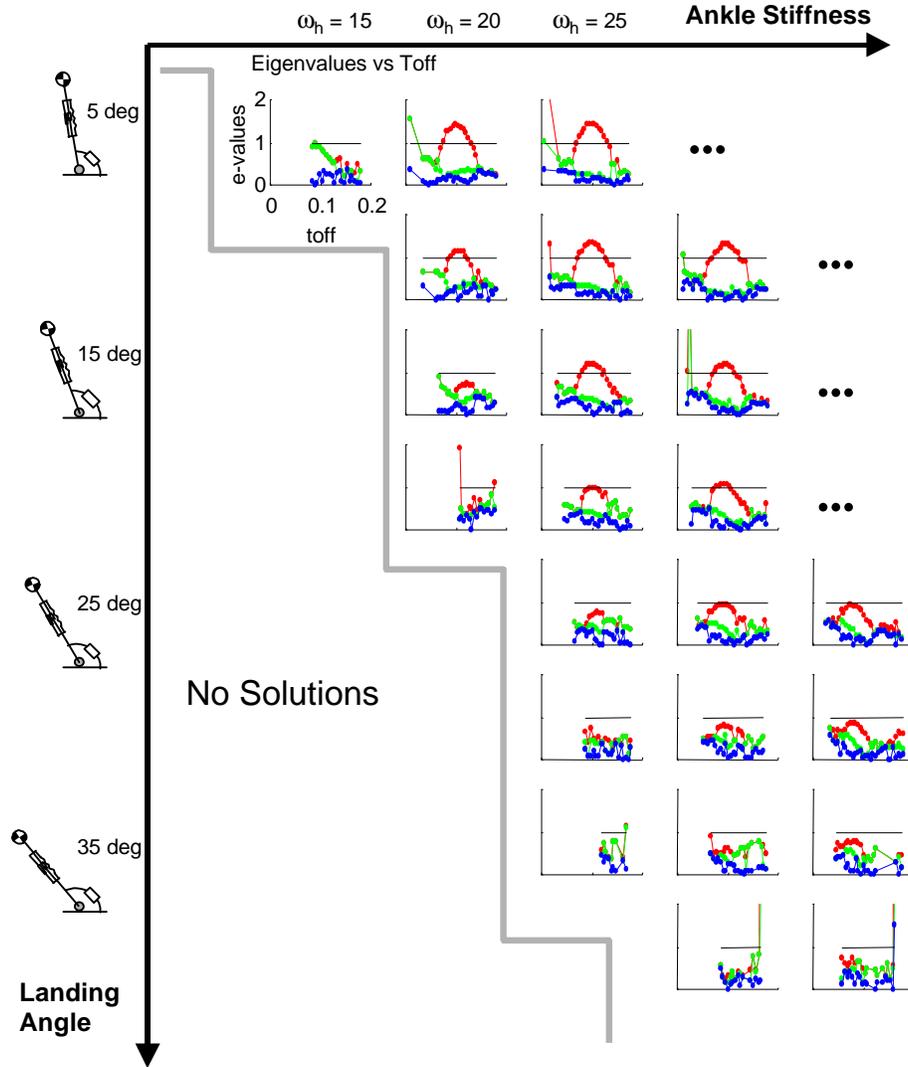
Figure 4-11 shows that speed for all combinations of ankle stiffness and leg angle is inversely proportional to  $t_{off}$ , such that steady-state "skittering" trajectories that initiate



**Figure 4-15.** Correlation between maximum work and maximum stride length. The stride period for which stride length is maximized over the continuum of solutions for a given combination of landing angle and ankle stiffness is plotted against the corresponding stride period for which work is maximized. Each line represents the period for a given leg landing angle.

thrust as soon as possible, and not at the leg's maximum compression, are fastest. Stride length, as shown in Figure 4-13 plotted against  $t_{off}$ , is not maximized for a minimum  $t_{off}$ . Rather, stride length is maximized in trajectories in which thrust is initiated near maximum compression of the leg. Similar to the basic SLIP case, skittering trajectories have shorter stride lengths and less net work performed per stride by the actuators, but their corresponding shorter periods, and consequently their higher stride frequency, result in overall greater forward speed.

Softer ankle stiffnesses and greater leg landing angles increase both the maximum speed and maximum stride length possible, with ankle stiffness having the more significant effect. This is due to the fact that, as ankle stiffness decreases and leg angle increases, the trajectories of the body tend to be more horizontal, and less vertical. For example, a stiff ankle spring and a nearly-vertical leg angle results in trajectories that are very similar to the trajectories of the vertical hopper described in the previous chapter. A consequence of this



**Figure 4-16.** Magnitude of the eigenvalues of the Jacobian of the Poincaré Map for the continuum of steady-state solutions, given a range of leg landing angles and ankle stiffnesses. As shown, the maximum magnitudes increase near low values of  $t_{off}$ , and near mid-range values of  $t_{off}$ , in which thrust is activated after maximum compression.

is that for more vertical trajectories, multiple solutions arise for a range of stride periods, similar to the vertical hopper. As the trajectories become more horizontal, the range of multiple solution decreases. However, as shown before, there is a limit in the space of ankle stiffness and landing angle for which solutions will exist. As a result, maximum speed and stride length are optimized in parameter combinations that lie at the edge of the region for which solutions exist (overall maximum speed and stride length occur at  $\omega_h=15$  and  $\theta_i=5\text{deg}$ , since  $\omega_h$  has the more significant effect). Experience with the Sprawl robots confirms that decreasing the stiffness of the hip flexures and angling the legs more aggres-

sively toward the back dramatically increases the speed of the robot, but that at these configurations the robot is more likely to crash and "nose-dive."

Figure 4-14 shows the work per stride performed by the actuator in each of the steady-state solutions for the range of ankle stiffness and leg landing angle combinations. There are several observations to be made. First, combinations of ankle stiffness and leg landing angle with the longest maximum stride lengths do not correspond to the combinations with the greatest maximum work. For example, for a leg landing angle of 15 degrees, the ankle stiffness for which steady-state solutions exhibit the greatest peak in work is  $\omega_h=30$ . But the ankle stiffness for which solutions exhibit the greatest peak in stride length is  $\omega_h=20$ . This shows that maximum work is not a good indicator to be used for optimizing either leg angles or ankle stiffness. However, for any particular combination of ankle stiffness and leg landing angle, stride length appears to be maximized by the value of  $t_{off}$ , and corresponding value of stride period, that simultaneously maximizes work. Figure 4-15 shows that the stride period for which stride length is maximized over the continuum of solutions for a given combination of landing angle and ankle stiffness is similar to the stride period for which work is maximized. This observation indicates that varying the configuration of the robot (ankle stiffness and leg landing angle) has a large impact on the resulting motion but that, for a particular configuration, the phasing of the actuator motor pattern must be properly "tuned" to achieve maximum performance.

Results of analyzing the stability of the steady-state trajectories described above are shown in Figure 4-16. The plots show the magnitudes of the eigenvalues of the Jacobian of the Poincare Map for each of the steady-state solutions. As shown, the open-loop system is stable for a range of system parameters, indicating that the ankle spring, which captures the effects of a sprawled posture and hip flexures, has a significant effect in increasing stability from the basic SLIP case. Two regions of instability are noted from the figures. The first is shown for small values of  $t_{off}$ , and it corresponds to the trajectories with maximum speed. Similarly to the basic SLIP, the eigenvector of this unstable mode can be shown to be largely in the direction of  $\Delta\theta_V$ . The second instability is shown for mid-range values of  $t_{off}$ . This instability can be shown to be in the direction of  $\Delta t_{off}$ , and corresponds to solutions in which thrust activation has started to occur after maximum compression, similar to

the instability in the "Long Thrust" vertical hopper. As shown, this instability is decreased for solutions with greater leg landing angle, as the trajectories become more horizontal and less vertical.

## 4.4 Chapter Conclusions

In this chapter, I analyzed two variations of a simplified planar running model. The first, a general "Spring-loaded Inverted Pendulum" or "basic SLIP," follows from traditional analysis of running systems as a general model for locomotion. The second, termed "Sprawl-type," is based on the specific mechanics of the Sprawl family of robots, and is intended to provide more direct insights into specific strategies that can be used for design and adaptation of the robots. The following conclusions are made from the analysis of the two models:

- In planar running, given a set of system parameters, there is a continuum of steady-state running trajectories that can be parameterized by  $t_{off}$ , the time after landing that thrust is activated. This observation expands the dynamic behavior of running systems beyond the traditional approach, as originated by Raibert, in which thrust is activated at maximum compression.
- Within this continuum of steady-state solutions, forward velocity is maximized in solutions with minimal  $t_{off}$ , termed "skittering," and not in solutions in which thrust is activated at maximum compression. Solutions in which  $t_{off}$  is minimal are able to support large horizontal landing velocities, in a motion that is similar to "vaulting" with a stiff pole. As a result of the minimal  $t_{off}$  and large horizontal motion, both stance and airborne phases have a short duration and solutions are associated with minimal stride periods. Thus, although stride length (length travelled per stride) is not optimal in these solutions, their associated high stride frequency results in maximal forward velocity.
- Stride length is optimized in steady-state solutions that are related to maximizing the work performed by the actuator during the stride. In these solutions, the timing of the thrust activation relative to the motion of the system is such that the net amount of

work is maximized. This situation may coincide with thrusting at maximum compression, if the thrust duration is "long enough" such that deactivation occurs after take-off.

- Under open-loop control, which implies that the leg landing angle is fixed and that thrust activation is controlled by a fixed motor pattern, stability can be decoupled into two general modes: stability in timing, or phasing, of the motor pattern, and stability in the angle of landing velocity. Instability in the timing of thrust activation, relative to the instant of landing, is due to the same mechanism as found in the analysis in the previous chapter for vertical hopping, and occurs in the range of solutions in which thrust is initiated after maximum compression of the leg, in the "Long Thrust" case.
- For the "basic SLIP," instability in the angle of the landing velocity makes the open-loop model unstable over the continuum for the range of parameters explored here. It was found that "skittering" solutions, or solutions with nearly horizontal landing velocities, are less unstable in this mode.
- As an aside, it was found that the unstable mode could theoretically be stabilized in a "partially open-loop" hopper by controlling the timing of thrust activation, as long as thrust activation occurred well before or well after maximum compression. In this hopper, leg landing angle is fixed, but thrust is activated by a timer that is reset at landing. This provides an alternative to Raibert's leg placement controller for stabilizing the basic SLIP.
- For the "Sprawl-type" model, stable behavior could be found for a range of system parameters. The instabilities in timing and landing velocity angle were decreased as the motion became more horizontal and less like a vertical hopper. Stumbling, or "nose-diving" behavior in the Sprawl robots is explained by the region of the parameter space for which no steady-state solutions exist (and not due to unstable steady-state solutions).
- For "Sprawl-type" running, it was found that system configuration, in terms of leg landing angle and ankle stiffness, plays a significant role in determining the overall character of the motion (whether the trajectories are more horizontal or more verti-

cal) and has a large effect on speed and stride length. Optimal configurations are those near the edge of the region in the parameter space for which there exist solutions, indicating a basic trade-off between performance and stability. Moreover, it was found that for a given leg landing angle and ankle stiffness, optimal stride length is still correlated to maximal work performed by the actuator per stride.

These observations suggest the following general procedure for optimizing the performance of a Sprawl-type robot:

- First, hip flexures should be made as compliant as possible and leg angles as great as possible a desired distance in the parameter space from the region in which no steady-state solutions exist, given a desired trade-off between performance and stability.
- Next, the stride period is chosen to either optimize for speed or stride length. To optimize for speed, the minimal stride period for which a solution exists should be chosen (or the minimal stride period that the actuator bandwidth will allow, whichever is lower). To optimize for stride length, the stride period for which work per stride is maximized should be chosen.

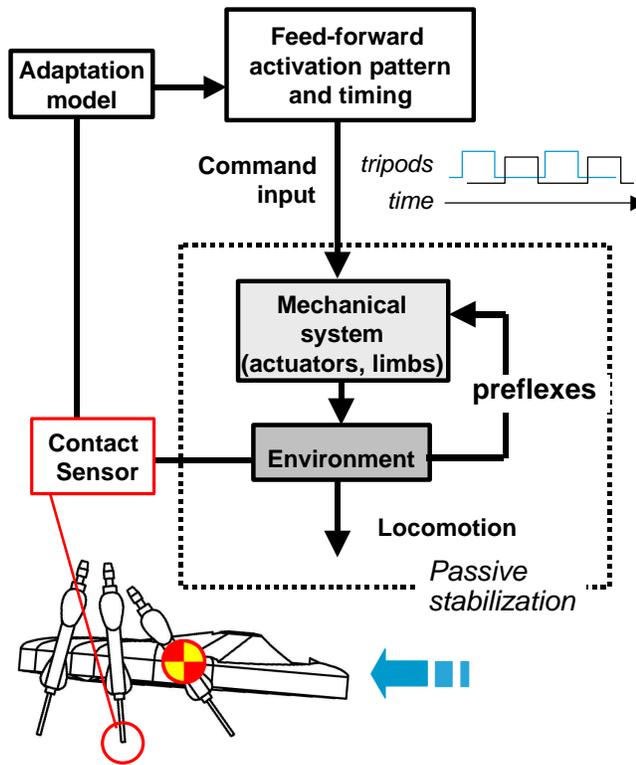
As previously suggested, this optimization, or "tuning," of the stride period could be performed online while running, requiring only the monitoring of thrust activation and deactivation relative to landing or take-off. This idea is tested experimentally in the following chapter.

# 5 Stride Period Adaptation for a Hexapedal Running Robot

In this chapter, a novel adaptation strategy for adjusting the stride period in a Sprawl hexapedal running robot is demonstrated. Experimental performance tests for the robot Sprawlita are first presented, which show that the underlying relationships in the dynamics of running systems established in previous chapters are evident in the more complex robot's running. A prototype adaptation scheme that is based on these observations is then described, and experiments in which the stride period is automatically adjusted from sub-optimal starting conditions are presented. These tests demonstrate that knowledge of the underlying dynamics in such a system can lead to stable adaptation strategies with minimal sensory requirements.

## 5.1 Stride Period Adaptation

The Sprawl family of robots (Cham et al., in press) demonstrates that a well-designed mechanical system with appropriate placement of compliance and damping and leg configuration can locomote quickly and robustly without sensory feedback. The passive mechanical properties of the system create the locomotion and reject disturbances to the steady state motion of the system, such that they can be controlled by an open-loop fixed motor pattern. However, as noted before, a particular open-loop motor pattern or set of leg angles may not always result in optimal running given changes in terrain (e.g. slope) or loading conditions (e.g. carrying an object). For example, as explained in Chapter 1, a stride period that is optimal for one ground slope may not be optimal for others. Thus, to increase the versatility of the robots, it is desired to introduce adaptation, in which the parameters of the open-loop system (leg angles and stride period and duty cycle of the motor pattern) are



**Figure 5-1.** Adaptation scheme in which rapid disturbances are rejected by the passive stabilizing properties of the mechanical system. Sensory information is used at a slower rate to adapt the parameters of the open-loop motor pattern in response to changes in the environment or loading conditions.

adjusted, or "tuned," to achieve an optimal desired mode of locomotion for the given environment or loading conditions.

An adaptation framework that complements the design of the robots and takes advantage of its self-stabilizing properties is as shown in Figure 5-1. The passive properties of the mechanical system are allowed to create the locomotion and reject rapid disturbances, while a slower feedback loop analyzes sensory information and adjusts the open-loop parameters to optimize for changes in the environment or loading conditions. This significantly reduces sensor bandwidth requirements, and allows for sensor failures, in which case the system simply reverts to its nominal open-loop behavior.

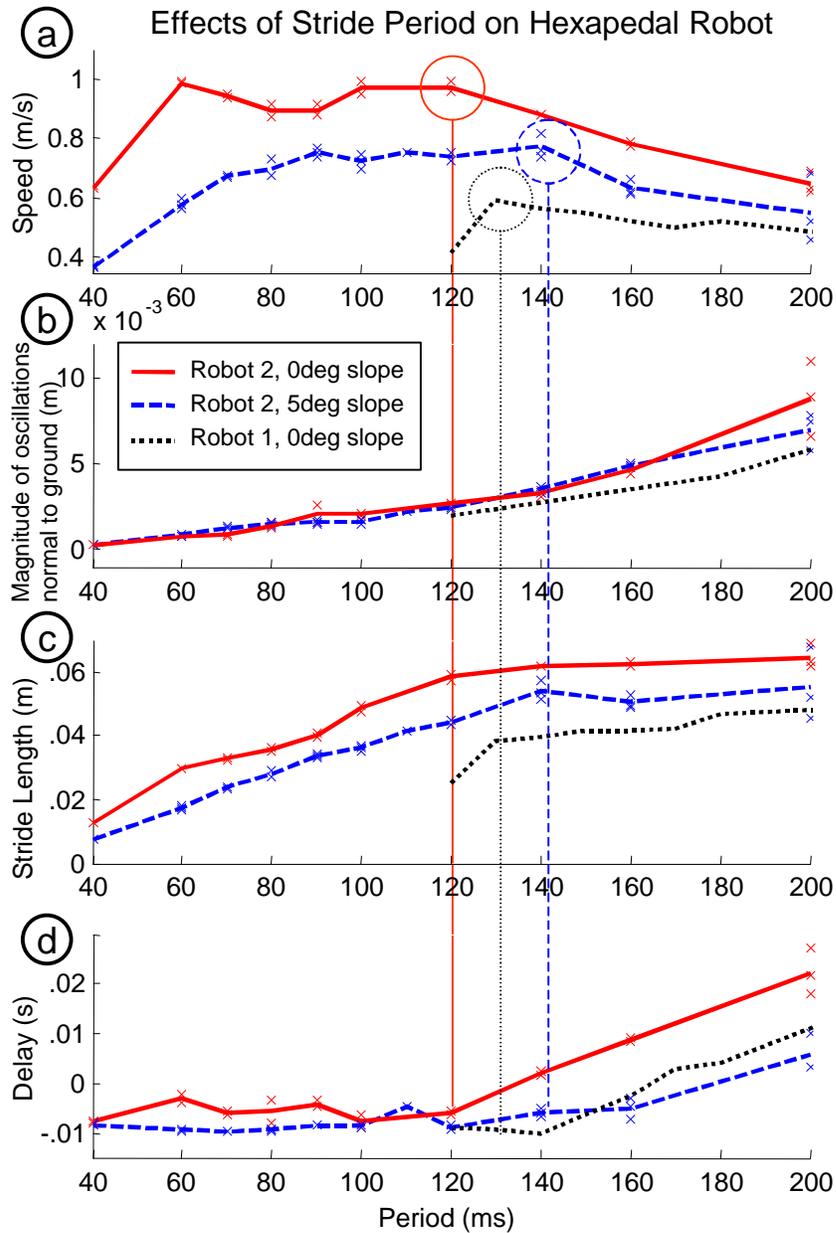
Sensor requirements can be further simplified by taking advantage of a priori knowledge of the system's dynamics. Knowledge of the basic mechanics of the system's locomotion can help identify which parameters to tune, and inform how to change them to achieve optimal behavior. Furthermore, such information could also allow quantities of

interest that are difficult to measure, such as velocity or net work, to be "observed" through other events that are known to be related by the system's dynamics. For example, the analysis in the previous chapters suggest that the amount of work performed by the actuator is related to the timing of thrust activation and deactivation relative to landing and take-off. Using this relationship, the adaptation scheme presented in this chapter adjusts the stride period using only information from a simple binary contact switch in one of the robot's feet.

In this chapter, we focus mainly on adaptation of the stride period of the open-loop motor pattern. One of the important results from the previous section is that, while changing the leg angles and hip flexures can have a significant effect on the motion of the system, it is possible to "tune" the stride period independently of these configuration parameters. In other words, it is possible to develop heuristics for the adaptation of the stride period that find optimal values, even if the optimal values change with leg angles and hip flexures. Assuming that instabilities are reduced for configurations of interest with more horizontal leg angles, as shown to be the case by Figure 4-16 for the Sprawl-type running model, these heuristics can focus only on maximizing performance. For example, in the case of maximizing forward velocity, previous analysis suggests a heuristic that chooses the shortest possible stride period, given the limitations of the actuator's bandwidth. In the case of maximizing stride length, the heuristic would choose the stride period which maximizes the net amount of work performed by the actuator on the system. Given this indication that leg angles and hip stiffness can be adapted separately from stride period, the adaptation of leg angles and hip stiffness is left for future development, and only stride period adaptation is demonstrated in this chapter.

## **5.2 Robot Performance Tests**

The vertical and planar running models provided insight into the basic behavior of an open-loop hopping system with passive properties in terms of the work performed by the actuator and the resulting performance. In order to develop an adaptation law for the six-legged, multi-DOF robot, we must look at the factors that affect its performance and see whether the same basic mechanisms are evident. Figure 8 shows the performance results of the hexapedal robot as a function of open-loop stride period for three different cases. The dotted lines represent the results for a first prototype, here called robot 1, running on flat



**Figure 5-2.** Experimental results of performance tests in Sprawlita, one of the Sprawl robots. The dotted line represents results for the robot with pneumatic actuators with significant damping. Solid lines represent results for the robot with pistons with less damping. Dashed lines represent the second robot running on a 5 degree sloped surface. As shown, despite the variation in robot parameters, the stride period for which stride length is maximized is correlated to a change in slope in the time difference between thrust deactivation and take-off.

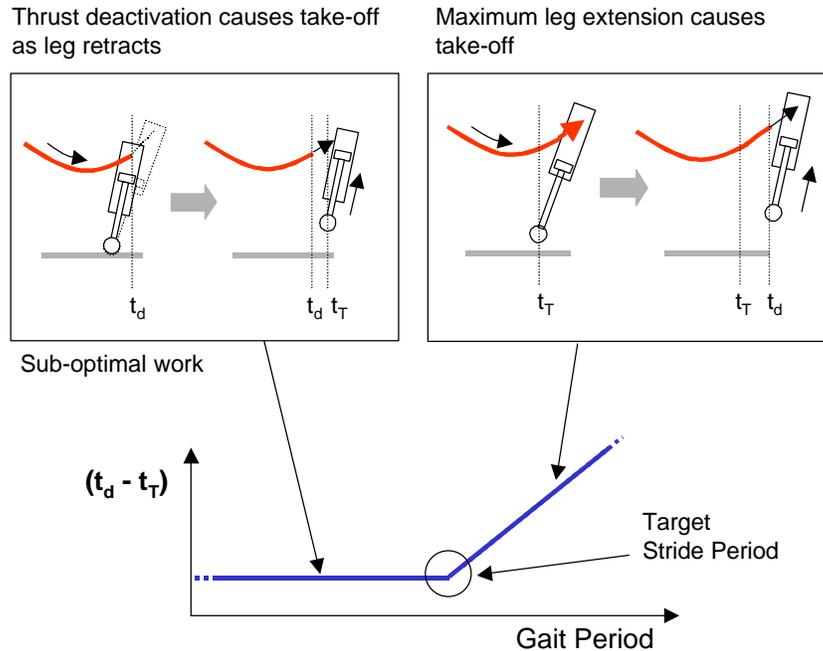
ground. The solid lines are for the same robot on flat ground, but with different actuators, here called robot 2. The new actuators are pneumatic pistons with faster air flow and less damping. Finally, the dashed lines are for robot 2 running on a 5 degree uphill slope.

As shown in Figure 5-2a, starting from high values of the stride period, forward speed increases as the stride period is decreased. At some point, speed starts to level off, reaches a maximum, and then decreases dramatically below a certain value of the stride period. This range, or "hump," over which speed is maximized occurs over different ranges for each of the three cases, again motivating the need for adaptation. The shape of this speed curve can be explained by the observations made in previous chapters on the simplified models.

For long periods, the motion of the robot was observed to be associated with "Hop-settle-fire" behavior similar to that observed in the one-DOF model, where the stride period is long enough that much of the energy from the previous hop is dissipated before thrust is initiated. Speed is increased as the period moves away from this behavior. The initial levelling off of the speed curve as stride period is decreased corresponds to the point in which stride length, plotted in Figure 5-2c, is no longer maximized. Although stride length is further decreased as stride period gets shorter, speed continues to be maximized, due to the fact that speed is the product of stride length and the inverse of stride period. However, speed dramatically decreases below a certain period, due to the limitations in bandwidth of the pneumatic actuators, which limits the maximum speed attainable.

Thus, similar to the simplified planar models, speed is not necessarily maximized at the same stride period as stride length. Instead, speed is maximized over a range of periods bounded on one side by the bandwidth of the actuators, and on the other side by the stride period in which stride length is maximized. As a goal of the adaptation, the period in which stride length is maximized was chosen. Although speed may be further maximized by shorter periods, the gain in speed is small relative to the increased energy consumption due to higher stride frequencies and lower efficiency. It remains to be verified whether, as predicted by the simplified models, stride length is maximized with maximum work performed by the actuators.

Since work performed by the actuator is difficult to measure directly, we observe it indirectly through measurements of thrust timing. In the case of the Sprawl robots, the legs do not have a spring along the length of the leg with a nominal length that determines when ground contact occurs. Instead, for a given leg, contact with the ground occurs when the



**Figure 5-3.** Illustration of indirectly measuring work performed by the actuator by measuring the time difference between take-off and actuator deactivation. Optimal work is done when the thrust is deactivated just as the leg reaches maximum extension. As shown, when thrust is deactivated before maximum extension, the time difference is a low constant value, as take-off occurs due to leg retraction. When thrust is deactivated well after maximum extension, the time difference is an increasing positive value.

pistons extend, shortly after the valves are activated. Thus, thrust activation and landing occur a fixed time delay apart. Take-off, or loss of contact with the ground, on the other hand, can occur when the leg reaches maximum extension, or when thrust is deactivated, which causes the leg to retract. As a result, we can indirectly measure the work performed by the actuator by looking at the time delay between the time that the valves are deactivated, and the time that take-off occurs. As indicated by the analysis of the simplified models, work is maximized when thrust deactivation occurs near the time that take-off occurs, given by the leg reaching maximum extension. Suboptimal work occurs when thrust is deactivated well before or well after maximum extension occurs. In situations in which thrust is deactivated well before maximum extension, loss of contact with the ground will be sensed a short fixed time after deactivation. In situations in which thrust is deactivated after maximum extension, then loss of contact will be measured before thrust deactivation. This time difference between  $t_d$ , the time that thrust is deactivated, and  $t_T$ , the time that take-off occurs, is illustrated in Figure 5-3.

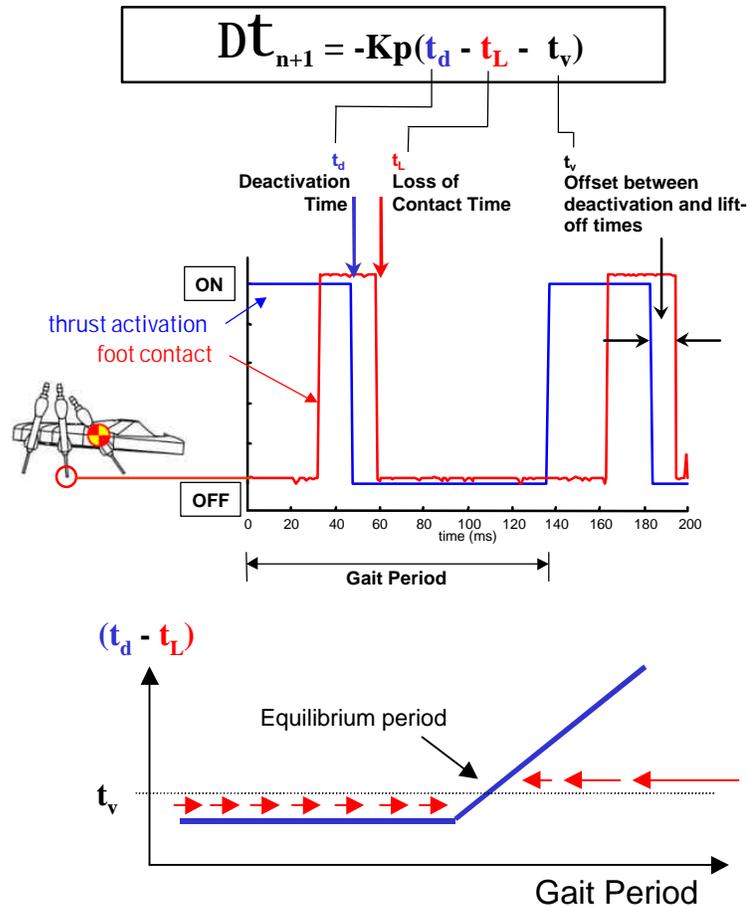
The time delay,  $(t_d - t_T)$  is plotted in Figure 5-2d for the different cases examined. This time delay,  $(t_d - t_T)$ , is positive for long stride periods, which indicates that thrust application ends after lift-off, here caused by end-of-stroke or full extension. This delay also monotonically increases for longer periods since thrust application, or  $t_{on}$ , is set as a fixed percentage, or duty cycle, of the stride period due to valve and air flow limitations. Below a certain range of stride periods, however, the time delay is a nearly constant small negative value. In effect, deactivation of the tripod causes the spring-loaded leg pistons to retract and lose contact with the ground before full extension.

This change in the slope of the time delay  $(t_d - t_T)$  relative to the stride period occurs near the period for which stride length begins to decrease and ground speed starts to level off. Although the dynamics of the robot's locomotion are affected by many factors, it is apparent that the stride period in which the amount of net positive work performed by the actuators is maximized, as indicated by the time delay  $(t_d - t_T)$ , has a first-order correlation with the period in which stride length is maximized. This correlation is used as the basis for the simple adaptation law described in the following section.

### 5.3 Adaptation Strategy

Previous results motivate the robot stride period adaptation strategy described here. As illustrated by the vertical and planar model, it is advantageous to use a stride period that results in a steady-state cycle in which thrust is deactivated near the point where full piston extension occurs in order to maximize work input. Similar to the simplified models, shorter stride periods result in sub-optimal work input as thrust is terminated before maximum extension. Moreover, as in the simplified models, longer stride periods result in suboptimal “Hop-settle-fire” behavior and sometimes in period-1 unstable oscillations. The experimental results in the previous sections confirm that these observations do characterize the behavior of the more complex robot. A prototype adaptation law for maximizing ground speed that takes these findings into consideration using foot contact information is as follows:

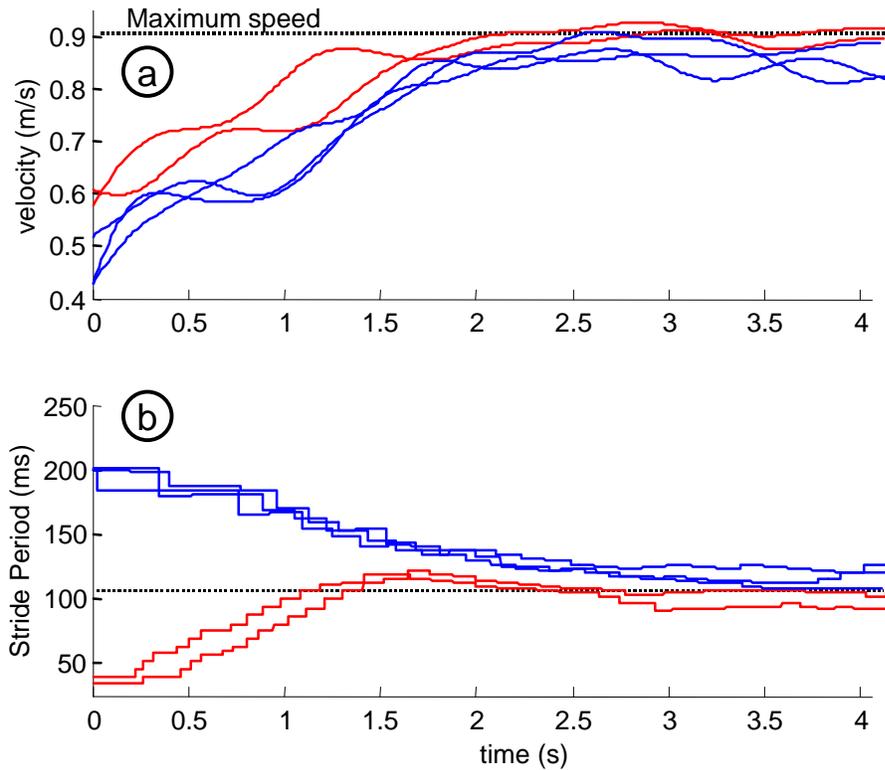
$$\tau_{n+1} = \tau_n - K_p(t_d - t_T - t_v) \quad (5.1)$$



**Figure 5-4.** Stride period adaptation strategy. The stride period is updated after every stride cycle based on the measured times that valve deactivation and loss of contact occur. As illustrated, the adaptation strategy seeks to find the period in which work is maximized, given by the period in which the time difference changes slope, and exceeds the threshold value  $t_v$ .

Here,  $K_p$  is the adaptation gain,  $t_v$  is a constant offset parameter,  $t_d$  is the time at which the valve is deactivated and  $t_L$  is the measured time that ground contact in the middle-foot is lost. Figure 5-4 illustrates what these quantities represent, where time is measured with respect to the initiation of the gait cycle, which starts when the valve for one of the tripods is activated. The adaptation law is based on contact information from a binary switch attached to the middle foot of the same tripod. The deactivation time  $t_d$  is determined by the stride period,  $\tau$ , and duty cycle, which in this case is specified as a fixed percentage of the stride period. If there is no measured ground contact information,  $t_L$ , then the period is not modified.

### Adaptation Results, 0deg slope, 80psi, Kp=0.5, Tv=6ms



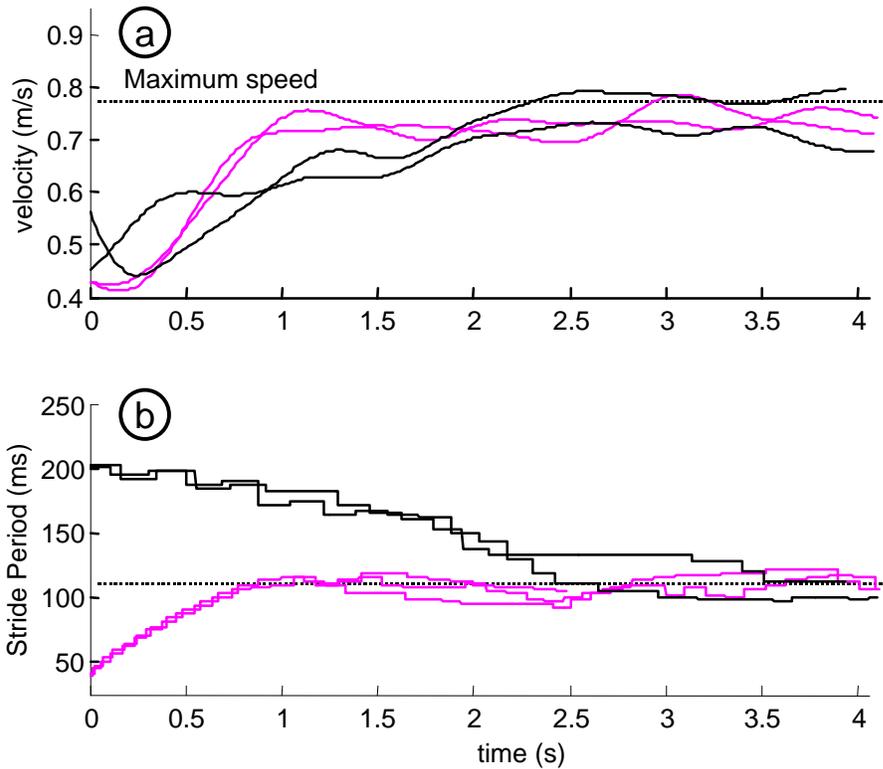
**Figure 5-5.** Adaptation results for flat terrain (dashed lines are approximate values for maximum speed established empirically). The figures show the ground speed of the robot and the stride period as it is adapted from suboptimal starting conditions.

Intuitively, this simple adaptation law can be described as trying to decrease the stride period as much as possible without exceeding the bandwidth of the actuators and without terminating the thrust application before full extension (to maximize available work). The stride period reaches an equilibrium value when  $\Delta\tau$  is zero, which occurs when  $(t_d - t_T)$  is equal to the offset value,  $t_v$ . Since the delay  $(t_d - t_T)$  is nearly constant for lower stride periods, the offset value  $t_v$  is adjusted slightly above zero, so that the equilibrium stride period coincides with the change in slope of the delay  $(t_d - t_T)$  with respect to the stride period (see Figure 5-4).

## 5.4 Adaptation Results

Figure 5-5 shows test results of the adaptation law implemented in the hexapedal robot 2 running on flat ground for several experiments in which the stride period was started at suboptimal values. Figure 5-5a shows the ground speed of the robot as a function of time, and

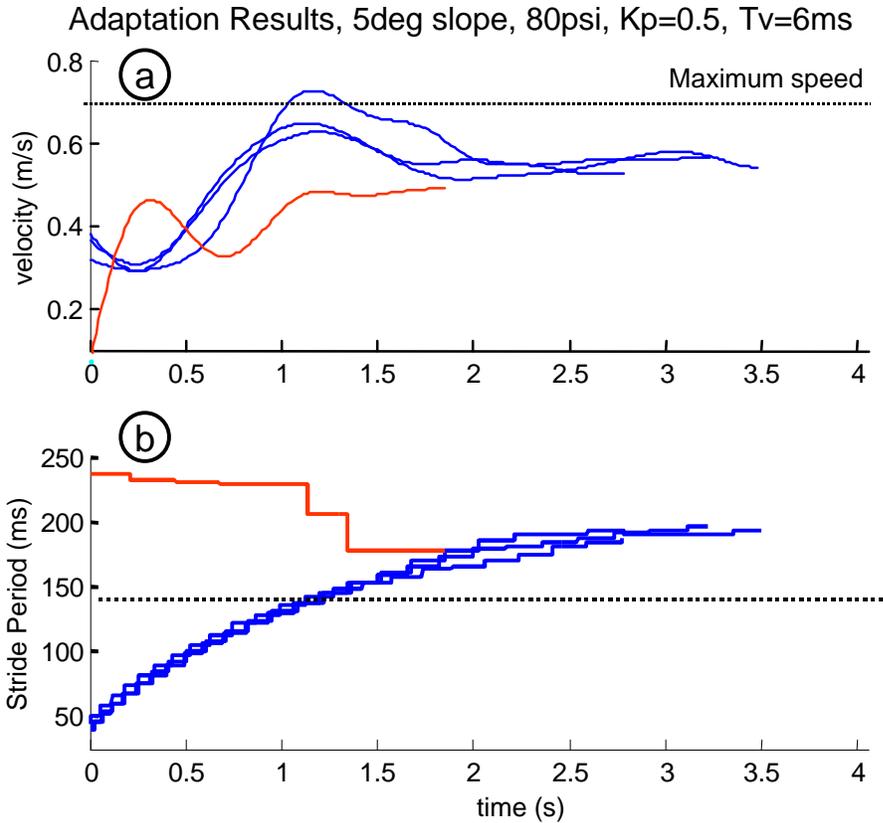
Adaptation Results, 0deg slope, 70psi,  $K_p=0.5$ ,  $T_v=6ms$



**Figure 5-6.** Adaptation results for flat terrain with a 13% decrease in pneumatic actuator input pressure. The adaptation optimizes ground speed by converging to a slightly higher stride period than for the lower input pressure.

Figure 5-5b shows the stride period after each stride cycle, or learning loop, in which  $t_T$  was measured. The gain  $K_p$  was experimentally chosen to give the adaptation a fast learning rate while still achieving convergence. Note from Figure 5-5b that, although only a simple contact switch was used, the measured values of  $t_T$  are still prone to some noise, due to ground imperfections or disturbances to the robot, and adaptation does not necessarily proceed smoothly. As shown, the adaptation causes the stride period to converge and to maximize speed. This adaptation strategy was also shown to maximize speed in robot 1, with different pneumatic pistons, and for the case where the input actuator pressure was decreased in robot 2 by 13% (shown in Figure 5-6).

For an uphill ground slope of 5 degrees, the adaptation strategy also converges to an equilibrium stride period, as shown in Figure 5-7. This new equilibrium period (~170ms) is higher than the equilibrium period for flat ground running (~110ms) and



**Figure 5-7.** Adaptation results for an uphill slope of 5 degrees. The adaptation strategy improves the locomotion, but converges to a stride period slightly higher than the optimal stride period.

results in faster uphill running than with the period that corresponded to maximum speed in flat ground. However, the new equilibrium period is somewhat higher than the period found to maximize speed at 5 deg. slopes (~140ms). This indicates that, although it works to improve locomotion speed when transitioning to sloped terrain, the simple threshold-based adaptation law implemented here results in errors in the equilibrium stride period for maximum speed in uphill running. This is mainly attributed to the gradual change in slope in the plot of  $(t_d - t_T)$  for 5 deg uphill terrain compared to the prominent “kink” in the corresponding plot for flat terrain (see Figure 5-2d). The use of a threshold to detect this change in slope results in equilibrium periods that are longer than the period of maximum speed. Future work will experiment with more sophisticated ways to detect this change in slope. Furthermore, the factors that affect uphill running may need to be re-examined. For example, in seeking to increase the stride frequency for optimal ground speed in flat terrain,

the prototype adaptation law presented here reduces oscillations in the direction normal to the ground, which we believe may have a significant role while climbing up-hill terrain.

## 5.5 Chapter Conclusions

An adaptation strategy for the stride period of an open-loop controlled hexapedal robot was demonstrated. The strategy is based on observations of the basic relationships in the dynamics of simplified models of hopping and running. In particular, the strategy seeks to maximize the stride length by maximizing the net work performed by the robot's actuators, which analysis of the simplified models suggests are simultaneously maximized, independently of the system's other parameters. The net work is indirectly measured by the timing of thrust deactivation and take-off, which previous analysis indicated must coincide in order for maximum work to be performed. The adaptation law uses only the sensed duration of ground contact during each stride, and was shown to cause the stride period to converge to values that maximize speed for a range of robot-to-robot variations and operating pressures. When making the transition from level to uphill running, the robots converge to slightly suboptimal values of stride period and velocity. The difficulty in this case is that the transition between optimal and over-long periods is less distinct and less easily identified with a simple threshold test. More sophisticated detection of the transition is an area of ongoing work.

More generally, the adaptation scheme presented in this paper is an example of an approach that is particularly well suited for small, biomimetic robots by requiring no expensive or sophisticated sensing or feedback. In this case, only binary switches are needed to provide an estimate of ground contact time. The adaptation scheme takes advantage of the passive properties of the robot that allow it to run stably over a range of open-loop stride frequencies and actuator duty cycles. In the event of sensor failure, the performance of the robot degrades only to that of the open-loop system without adaptation. This approach allows the robots to remain simple, inexpensive and robust while also being able to "tune" themselves to accommodate individual variabilities and changes in operating conditions.

Future work will build upon the simple adaptation law tested in this paper to incorporate other simple sensor information (e.g. inclinometer, contact switches in other feet) in

order to increase performance and adaptability. As discussed previously, further understanding of the robot's dynamic interaction with different types of terrain such as sloped or compliant surfaces will allow us to increase the adaptation's versatility.

# 6 Conclusions

Fast and robust locomotion is possible without sensory feedback. The Sprawl family of robots have demonstrated that a simple mechanical system with properly designed passive properties can be controlled open-loop to achieve significant speed and obstacle clearance. This thesis has investigated the basic relationships among performance, stability and the open-loop motor pattern in running systems like the Sprawl robots. The approach taken has used analytical, numerical and experimental methods in studying open-loop running systems starting with an analytically-tractable simplified vertical hopper, followed by simplified planar running models, and finally experimental data of one of the Sprawl robots. The main conclusions of this thesis are in the form of generalizations of the factors that determine stability and performance, the trade-offs between the two, and the inherent limitations of open-loop control for running systems.

## 6.1 Main Conclusions

Performance, in terms of hopping height for vertical hopping and stride length and forward velocity for planar running, was found to be well characterized by the net amount of work performed by the actuator on the system during a stride. Stride periods of the open-loop motor pattern that result in steady-state trajectories in which maximum work is performed correspond to stride periods that maximize hopping height and stride length. The net amount of work, in turn, can be characterized by the timing of actuation activation and deactivation relative to the motion of the system. It was found that work is maximized for a single leg if actuation deactivation occurs near the point where the leg reaches maximum extension and, if possible given limitations in actuation duration, activation occurs near the leg's maximum compression.

Exploration of the timing of actuation activation led to an evaluation of running trajectories not previously considered by traditional analyses of running systems. Traditional

analyses commonly assume either conservative behavior (without energy loss), or actuation activation according to the control law implemented by Raibert (1986) in his hoppers, which initiates thrust when the leg is sensed to be at maximum compression. A significant finding is that activating thrust at maximum compression is not always optimal in terms of performance, given constraints in actuator duration. Exploring trajectories in which thrust is activated at other points in the cycle revealed trajectories with higher hopping heights, in the case of vertical hopping, and higher forward velocity in the case of the planar running. As an aside, the exploration of the timing of thrust actuation also led to the proposition of a novel strategy for stabilizing a running monopod, which was found to be unstable under open-loop control. This proposed strategy adjusts the time of thrust activation to correct for errors in the velocity at landing, allowing a monopod to have a fixed landing leg angle.

Open-loop running results in unstable trajectories when the system fails to re-synchronize its motion to the timing of the open-loop motor pattern after a perturbation. The previously established correlation between the amount of work performed on the system by the actuator to the timing of activation and deactivation helps explain this de-synchronization. Disturbances in the activation time cause a change in the work performed by the actuator that translates to changes in the duration of the time spend mid-air between strides. A steady-state trajectory is unstable if the resulting change in the airborne time acts to exacerbate the error in synchronization. Analytical and simulation results brought attention to two separate instances of this unstable behavior. The first affects situations in which thrust is deactivated after take-off, and occurs when thrust is activated shortly after maximum compression. The second occurs when thrust duration is short and the motor pattern frequency is high, and appears to result in period-2 behavior.

Examination of the trade-offs in stability and performance for open-loop running systems showed that optimal steady-state trajectories lie at the edge of stable behavior. In vertical hopping under open-loop control, optimal steady-state trajectories are near unstable trajectories. In addition, these trajectories can share stride periods with other undesirable trajectories, causing a loss of control over the resulting behavior. In planar running, optimal configurations of the system, in terms of leg angles and hip stiffnesses, are at the border of the region of the parameter space for which steady-state solutions exist. The main

consequence of these observations is that an open-loop running system may have to be operated at sub-optimal settings in order to avoid unstable or undesirable behavior.

Insights into the underlying dynamics can inform strategies for the adaptation, or "tuning," of the open-loop parameters. Such insights can also minimize the sensory information needed by adaptation strategies by relating performance to more easily measurable variables or events. A novel adaptation scheme is demonstrated in the hexapedal robots that adjusts the period of the open-loop motor pattern in response to changing conditions. The adaptation strategy takes advantage of the established relationships between the stride period, ground contact and performance gained throughout this work, and requires only binary contact information from one of the robot's feet.

## **6.2 Suggestions for Future Work**

In the control framework of the Sprawl robots, a feedforward controller activates the actuators according to a motor pattern signal, relying on the mechanical system to achieve locomotion and reject disturbances through "preflexes." Sensory information is used only to modify the parameters of the motor pattern, such as the period, at a much slower rate. This approach guarantees a stable baseline behavior in the event of sensor failure or noise due to uneven terrain or adverse environmental conditions.

Although open-loop stable steady-state trajectories have been found for statically-unstable configurations such as a monopod with a pointed foot (Mombaur, 2001), they are unlikely to share the same basin of attraction, that is, the range of allowable disturbances, as the Sprawl robots. In the case of the Sprawl robots, a sprawled posture and low center of gravity contribute greatly to their ability to sustain large perturbations during running. Pure open-loop control is unlikely to be practical for less stable running configurations such as the monopod. Thus, incorporating "reflexes" into such systems while still taking advantage of passive mechanical properties is an area of future work. In this case, finding open-loop stable trajectories for such configurations can help reduce the control requirements, in terms of bandwidth and actuator authority.

The introduction of reflexes in an open-loop-preflex scheme that directly affects the motor commands presents interesting possibilities. In studies of biological systems, it has

been found that sensory feedback modifies the signals from Central Pattern Generators in more than just magnitude and frequency (Abbas and Full, 2000). Even the introduction of simple ground contact sensing for the activation of thrust in running can have a significant effect, as shown in this thesis. A controller which activates thrust a fixed time (called  $t_{off}$  in the analysis) after landing avoids some of the limitations of open-loop control, in terms of multiple solutions, and can be used to stabilize an unstable Spring-Loaded Inverted Pendulum. A control scheme which uses feedback control, but which is able to rely on mechanical properties and an open-loop pattern in case of sensor failure, can have many benefits.

Another area of future work involves adding other dynamic behaviors to the Sprawl family of robots such as rapid turning and navigation. Turning in the Sprawl robots has been shown to be effected by changes in the open-loop parameters such as leg angles and the timing of thrust activation. An interesting approach combines the use of simplified models such as the ones presented in this thesis and the use of numerical parameter optimization techniques. The use of both approaches would allow us to understand the underlying dynamics while exploring the system's full parameter space. In this case, the two approaches can inform and help guide one another with their results. For turning, it remains to be seen whether sensory feedback that causes a reflex reaction within a stride period can result in more efficient turning for increased maneuverability.

# Appendix

## Appendix A - Jacobian Matrices for Hybrid Linear Systems with Arbitrary Mode Boundaries

This section of the appendix derives a general formula for the Jacobian of the map between two arbitrary mode boundaries along a nominal trajectory for a class of Hybrid linear systems. This Jacobian maps disturbances about the nominal trajectory from one boundary to the other. It is assumed that the nominal trajectory is given, that it intersects the two given mode boundaries, and that it does so orthogonally to the boundary. The system is assumed to be linear and time-invariant between the two mode boundaries, and given by the following differential equations:

$$\dot{Z} = AZ + B \quad (\text{A.1})$$

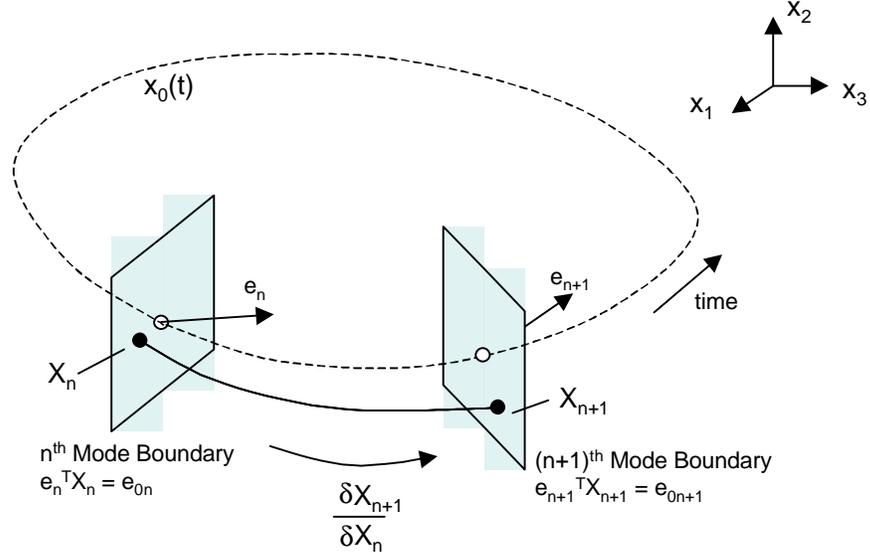
where  $Z$  is the physical state-space of the linear system, and  $A$  is assumed to be invertible. Because the mode transitions can be time-based (e.g. the activation of thrust due to the open-loop motor pattern), we augment the state of the system with  $t$ , a proxy variable used to represent time:

$$X = \begin{bmatrix} t \\ Z \end{bmatrix} \quad (\text{A.2})$$

Thus, the differential equations become:

$$\dot{X} = \begin{bmatrix} \dot{t} \\ \dot{Z} \end{bmatrix} = \begin{bmatrix} 0 & 0 \\ 0 & A \end{bmatrix} X + \begin{bmatrix} 1 \\ B \end{bmatrix} \quad (\text{A.3})$$

Note that "0" in the expressions is used in this derivation to represent both the scalar number zero, and zero matrices with numbers of rows and columns as needed to maintain consistency. The number "1" always represents the scalar number one.



**Figure A-1.** Illustration of Jacobian of map between two mode boundaries.

Illustrated in Figure A-1, the mode boundaries for which the Jacobian is derived are planar surfaces in the state space, represented by the following constraint equation:

$$e^T X = e_0 \quad (\text{A.4})$$

where  $e$  is the unit vector normal to the plane, and  $e_0$  is a scalar constant. The orthogonality assumption requires that the derivative of the state at the intersection with the boundary is not orthogonal to the unit vector normal to the plane:

$$e^T \dot{X}_{boundary} \neq 0 \quad (\text{A.5})$$

Intuitively, this assumption ensures that the boundary is normal to the trajectory, such that small changes in the boundary or in the trajectory will not result in large changes in the results of the analysis. The implications of violating this assumption are made clear later in this derivation.

Since the time at the end of the mode is given by,

$$t_{n+1} = t_n + t_\Delta \quad (\text{A.6})$$

where  $t_\Delta$  is the scalar time duration of the trajectory between the two mode boundaries, the time solution of the system within the mode is given by:

$$X_{n+1} = \begin{bmatrix} 1 & 0 \\ 0 & e^{At_\Delta} \end{bmatrix} (X_n - X_e) + X_e + \begin{bmatrix} t_\Delta \\ 0 \end{bmatrix} \quad (\text{A.7})$$

where  $X_{n+1}$  and  $X_n$  are the states at the two mode boundaries,  $e^{At_\Delta}$  is the matrix exponential, and  $X_e$  is the augmented equilibrium state of the linear system, given by:

$$X_e = \begin{bmatrix} 0 \\ -A^{-1}B \end{bmatrix} \quad (\text{A.8})$$

The above time solution "maps" the state from one boundary to the next. To find the Jacobian of this map, we take the derivative with respect to the initial state,  $X_n$ , taking advantage of the derivative properties of the matrix exponential:

$$\frac{\partial e^{Af(s)}}{\partial s} = A e^{Af(s)} \frac{\partial f}{\partial s} = e^{Af(s)} A \frac{\partial f}{\partial s} \quad (\text{A.9})$$

resulting in:

$$\frac{\partial X_{n+1}}{\partial X_n} = \begin{bmatrix} 0 & 0 \\ 0 & A e^{At_\Delta} \end{bmatrix} (X_n - X_e) \frac{\partial t_\Delta}{\partial X_n} + \begin{bmatrix} 1 & 0 \\ 0 & e^{At_\Delta} \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \frac{\partial t_\Delta}{\partial X_n} \quad (\text{A.10})$$

This expression can be rewritten as:

$$\frac{\partial X_{n+1}}{\partial X_n} = \dot{X}_{n+1} \frac{\partial t_\Delta}{\partial X_n} + \begin{bmatrix} 1 & 0 \\ 0 & e^{At_\Delta} \end{bmatrix} \quad (\text{A.11})$$

where  $\dot{X}_{n+1}$  is the rate of change of the state at the second mode boundary. To show this, we start by grouping together the terms in Equation A.10 with  $(\partial t_\Delta)/(\partial X_n)$ , resulting in:

$$\frac{\partial X_{n+1}}{\partial X_n} = \left( \begin{bmatrix} 0 & 0 \\ 0 & A e^{At_\Delta} \end{bmatrix} (X_n - X_e) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) \frac{\partial t_\Delta}{\partial X_n} + \begin{bmatrix} 1 & 0 \\ 0 & e^{At_\Delta} \end{bmatrix} \quad (\text{A.12})$$

The first product of the first term can be rewritten as:

$$\begin{bmatrix} 0 & 0 \\ 0 & Ae^{At_\Delta} \end{bmatrix} (X_n - X_e) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ Ae^{At_\Delta}(Z_n - Z_e) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \quad (\text{A.13})$$

where  $Z_n$  is the physical state-space at the first mode boundary. We can rewrite this term by noting that,

$$Ae^{At_\Delta}(Z_n - Z_e) = A(Z_{n+1} - Z_e) = \dot{Z}_{n+1} \quad (\text{A.14})$$

Resulting in:

$$\begin{bmatrix} 0 \\ Ae^{At_\Delta}(Z_n - Z_e) \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 0 \\ \dot{Z}_{n+1} \end{bmatrix} + \begin{bmatrix} 1 \\ 0 \end{bmatrix} = \begin{bmatrix} 1 \\ \dot{Z}_{n+1} \end{bmatrix} = \dot{X}_{n+1} \quad (\text{A.15})$$

and,

$$\frac{\partial X_{n+1}}{\partial X_n} = \left( \begin{bmatrix} 0 & 0 \\ 0 & Ae^{At_\Delta} \end{bmatrix} (X_n - X_e) + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \right) \frac{\partial t_\Delta}{\partial X_n} + \begin{bmatrix} 1 & 0 \\ 0 & e^{At_\Delta} \end{bmatrix} = \dot{X}_{n+1} \frac{\partial t_\Delta}{\partial X_n} + \begin{bmatrix} 1 & 0 \\ 0 & e^{At_\Delta} \end{bmatrix} \quad (\text{A.16})$$

To find the term  $(\partial t_\Delta)/(\partial X_n)$ , we first make use of the constraint equation at the second boundary:

$$e_{n+1}^T X_{n+1} = e_{0(n+1)} \quad (\text{A.17})$$

and then, noting that taking the derivative with respect to  $X_n$  results in:

$$\frac{\partial}{\partial X_n} (e_{n+1}^T X_{n+1}) = e_{n+1}^T \frac{\partial X_{n+1}}{\partial X_n} = \frac{\partial e_{0(n+1)}}{\partial X_n} = 0 \quad (\text{A.18})$$

we can pre-multiply Equation A.16 by  $e_{n+1}^T$ , to yield:

$$e_{n+1}^T \frac{\partial X_{n+1}}{\partial X_n} = e_{n+1}^T \left( \dot{X}_{n+1} \frac{\partial t_\Delta}{\partial X_n} + \begin{bmatrix} 1 & 0 \\ 0 & e^{At_\Delta} \end{bmatrix} \right) = 0 \quad (\text{A.19})$$

This gives us a way to solve for  $(\partial t_\Delta)/(\partial X_n)$ :

$$e_{n+1}^T \left( \dot{X}_{n+1} \frac{\partial t_\Delta}{\partial X_n} + \begin{bmatrix} 1 & 0 \\ 0 & e^{At_\Delta} \end{bmatrix} \right) = 0 \quad (\text{A.20})$$

$$e_{n+1}^T \dot{X}_{n+1} \frac{\partial t_\Delta}{\partial X_n} + e_{n+1}^T \begin{bmatrix} 1 & 0 \\ 0 & e^{At_\Delta} \end{bmatrix} = 0 \quad (\text{A.21})$$

$$\frac{\partial t_\Delta}{\partial X_n} = -(e_{n+1}^T \dot{X}_{n+1})^{-1} e_{n+1}^T \begin{bmatrix} 1 & 0 \\ 0 & e^{At_\Delta} \end{bmatrix} \quad (\text{A.22})$$

The condition for orthogonality ensures that  $(e_{n+1}^T \dot{X}_{n+1})^{-1}$  is finite. Thus, the Jacobian can be re-written by substituting Equation A.22 into Equation A.16, resulting in:

$$\frac{\partial X_{n+1}}{\partial X_n} = -\dot{X}_{n+1} (e_{n+1}^T \dot{X}_{n+1})^{-1} e_{n+1}^T \begin{bmatrix} 1 & 0 \\ 0 & e^{At_\Delta} \end{bmatrix} + \begin{bmatrix} 1 & 0 \\ 0 & e^{At_\Delta} \end{bmatrix} \quad (\text{A.23})$$

We can further simplify by factorization:

$$\frac{\partial X_{n+1}}{\partial X_n} = -(e_{n+1}^T \dot{X}_{n+1})^{-1} (\dot{X}_{n+1} e_{n+1}^T - e_{n+1}^T \dot{X}_{n+1} I) \begin{bmatrix} 1 & 0 \\ 0 & e^{At_\Delta} \end{bmatrix} \quad (\text{A.24})$$

where  $I$  is the identity matrix. This simple formula allows calculation of the Jacobian of the map between two mode boundaries along a nominal trajectory. Having defined the nominal trajectory, the above expression is evaluated at the trajectory by substituting the values of  $A$  and  $e_{n+1}$  that correspond to the given mode and mode boundary, and the values of  $X_{n+1}$ , and  $t_\Delta$  that correspond to that nominal trajectory.

For a steady-state trajectory that undergoes several mode changes, assuming continuity of state at the boundaries, the overall Jacobian is given by the product of the individual Jacobians for each mode, starting from the boundary that defines the Poincare Section of the Poincare Map. However, any disturbance introduced at the Poincare Map must lie in the plane defined by the Poincare Section. As a consequence, if the formula in Equation A.24 is used, the overall Jacobian must be post-multiplied by a projection matrix  $P$  that projects disturbances unto this plane:

$$P = (I - e_{PS}e_{PS}^T) \quad (\text{A.25})$$

where  $e_{PS}$  is the unit vector normal to the Poincare Section. An eigenvalue of 0 is then expected, which can be ignored. An alternative approach is to "reduce" the state space at each mode boundary, if each of the boundaries constrains a particular state-space variable. In this case, the appropriate row and column is removed from the Jacobian in Equation A.24 that correspond to the variables constrained at each of the mode boundaries. In the case of the vertical hopper, mode boundaries are either purely time-based (activation and deactivation due to the open-loop motor pattern), or purely position-based (take-off and landing), such that reducing the state space is the approach taken.

## Appendix B - Derivation of Jacobian Matrices for the Vertical Hopper Model

This section of the appendix derives the individual Jacobians for the case of the vertical hopper. Although the formula derived in Appendix A could be used, in this Appendix we explicitly derive each of the elements of the individual sensitivity matrices that map disturbances between the reduced state variables at each mode transition. This explicit derivation is included as it provides some insight into the nature of the resulting expressions.

We first derive the Jacobian for the GROUND\_OFF mode that occurs right after landing and before thrust is initiated, which is defined as:

$$\frac{\partial \tilde{X}_N}{\partial \tilde{X}_L} = \begin{bmatrix} \frac{\partial \tilde{X}_N}{\partial t_L} & \frac{\partial \tilde{X}_N}{\partial \dot{y}_L} \end{bmatrix} \quad (\text{B.1})$$

This mode maps the state from a state-based mode transition event (landing) to a time-based event (thrust activation). We start by writing an expression for the state at thrust activation:

$$X_N = e^{At_{off}}(X_L - X_{eoff}) + X_{eoff} \quad (\text{B.2})$$

Taking advantage of the derivative properties of the matrix exponential,

$$\frac{\partial e^{Af(s)}}{\partial s} = A e^{Af(s)} \frac{\partial f}{\partial s} = e^{Af(s)} A \frac{\partial f}{\partial s} \quad (\text{B.3})$$

we then find the derivative of  $X_N$  with respect to the time variable  $t_L$ , the time within the motor pattern cycle at the instant of landing:

$$\frac{\partial X_N}{\partial t_L} = e^{At_{off}} A (X_L + X_{eoff}) \frac{\partial t_{off}}{\partial t_L} \quad (\text{B.4})$$

The time at landing and the time duration  $t_{off}$  are related by:

$$\begin{aligned} t_{off} &= \tau - t_L \\ \frac{\partial t_{off}}{\partial t_L} &= -1 \end{aligned} \quad (\text{B.5})$$

Expanding  $A$  from Equation B.2, and using,

$$X_L = \begin{bmatrix} 0 \\ \dot{y}_L \end{bmatrix} = \begin{bmatrix} 0 \\ -\dot{y}_T \end{bmatrix} \quad (\text{B.6})$$

we simplify Equation B.4 as:

$$\frac{\partial X_N}{\partial t_L} = e^{At_{off}} \begin{bmatrix} \dot{y}_T \\ 1 - 2\zeta\omega\dot{y}_T \end{bmatrix} \quad (\text{B.7})$$

Finally, we find the derivative of  $X_N$  with respect to the landing velocity, which is found from Equation 2:

$$\frac{\partial X_N}{\partial \dot{y}_L} = e^{At_{off}} \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad (\text{B.8})$$

Putting together these two column vectors, and factoring the matrix exponential, the final Jacobian is found to be:

$$\frac{\partial \tilde{X}_N}{\partial \tilde{X}_L} = e^{At_{off}} \begin{bmatrix} \dot{y}_T & 0 \\ 1 - 2\zeta\omega\dot{y}_T & 1 \end{bmatrix} \quad (\text{B.9})$$

The Jacobian for the airborne modes is found by the following simple relationships between the states at take-off and landing:

$$\begin{aligned} \dot{y}_L &= -\dot{y}_T \\ t_L &= t_T + 2\dot{y}_L \end{aligned} \quad (\text{B.10})$$

Thus, the Jacobian for the airborne mode is:

$$\frac{\partial \tilde{X}_L}{\partial \tilde{X}_T} = \begin{bmatrix} 1 & 2 \\ 0 & -1 \end{bmatrix} \quad (\text{B.11})$$

At this point we must treat the "Long Thrust" and "Short Thrust" cases separately for the rest of the individual Jacobians. For the "Long Thrust" case, the Jacobian for the GROUND\_ON mode is given by:

$$\frac{\partial \tilde{X}_T}{\partial \tilde{X}_N} = \begin{bmatrix} \frac{\partial t_L}{\partial X_N} \\ \frac{\partial \dot{y}_L}{\partial X_N} \end{bmatrix} \quad (\text{B.12})$$

To find these derivatives, we start with an expression for the state at take-off:

$$X_T = e^{At_{on}}(X_N - X_{eon}) + X_{eon} \quad (\text{B.13})$$

and we find the derivative with respect to  $X_N$ :

$$\frac{\partial X_T}{\partial X_N} = A e^{At_{on}}(X_N - X_{eon}) \frac{\partial t_{on}}{\partial X_N} + e^{At_{on}} \quad (\text{B.14})$$

using Equation B.13, we simplify this expression:

$$\frac{\partial X_T}{\partial X_N} = A(X_T - X_{eon}) \frac{\partial t_{on}}{\partial X_N} + e^{At_{on}} \quad (\text{B.15})$$

Since  $t_{on}=t_T$  and,

$$X_T = \begin{bmatrix} 0 \\ \dot{y}_T \end{bmatrix} \quad (\text{B.16})$$

we can solve for the first row of Equation B.12 by pre-multiplying Equation B.15 by the row vector  $[1 \ 0]$ :

$$[1 \ 0] \frac{\partial X_T}{\partial X_N} = [1 \ 0] \left[ A(X_T - X_{eon}) \frac{\partial t_T}{\partial X_N} + e^{At_{on}} \right] = 0 \quad (\text{B.17})$$

and we can solve for the derivative of  $t_T$  with respect to the state  $X_N$ :

$$\frac{\partial t_T}{\partial X_N} = - \left[ [1 \ 0] A(X_T - X_{eon}) \right]^{-1} [1 \ 0] e^{At_{on}} \quad (\text{B.18})$$

Expanding  $A$  and using Equation B.16, we find that:

$$\frac{\partial t_T}{\partial X_N} = -(1/\dot{y}_T) [1 \ 0] e^{At_{on}} \quad (\text{B.19})$$

To find the second row of Equation B.12, we premultiply Equation B.15 by  $[0 \ 1]$ , resulting in:

$$[0 \ 1] \frac{\partial X_T}{\partial X_N} = [0 \ 1] \left[ A(X_T - X_{eon}) \frac{\partial t_T}{\partial X_N} + e^{At_{on}} \right] = \frac{\partial \dot{y}_T}{\partial X_N} \quad (\text{B.20})$$

Thus, we can write the derivative for the GROUND\_ON mode in the "Long Thrust" case as:

$$\frac{\partial \tilde{X}_T}{\partial \tilde{X}_N} = \left[ \begin{array}{c} -(1/\dot{y}_T) [1 \ 0] \\ [0 \ 1] \left[ A(X_T - X_{eon}) (-(1/\dot{y}_T) [1 \ 0]) + I \right] \end{array} \right] e^{At_{on}} \quad (\text{B.21})$$

For the "Short Thrust" case, we first look at the active thrust mode, GROUND\_ON, which has a constant duration of  $t_{on}$ . The state at deactivation is given by:

$$X_D = e^{At_{on}}(X_N - X_{eon}) + X_{eon} \quad (\text{B.22})$$

Since  $t_{on}$  is constant, we can find the derivative with respect to  $X_N$  easily as:

$$\frac{\partial X_D}{\partial X_N} = e^{At_{on}} \quad (\text{B.23})$$

Finally, finding the derivative of the state at take-off with respect to the state at thrust deactivation follows similarly to the derivation of Equation B.21. This derivative is given by:

$$\frac{\partial \tilde{X}_T}{\partial \tilde{X}_D} = \begin{bmatrix} \frac{\partial t_L}{\partial X_D} \\ \frac{\partial \dot{y}_L}{\partial X_D} \end{bmatrix} \quad (\text{B.24})$$

To find these derivatives, we start with an expression for the state at take-off:

$$X_T = e^{At_{off2}}(X_D - X_{eoff}) + X_{eoff} \quad (\text{B.25})$$

where  $t_{off2}$  is the duration of this mode. We find the derivative with respect to  $X_N$ :

$$\frac{\partial X_T}{\partial X_D} = A e^{At_{off2}}(X_D - X_{eoff}) \frac{\partial t_{off2}}{\partial X_D} + e^{At_{off2}} \quad (\text{B.26})$$

and using Equation B.25, we simplify this expression:

$$\frac{\partial X_T}{\partial X_D} = A(X_T - X_{eoff}) \frac{\partial t_{off2}}{\partial X_D} + e^{At_{off2}} \quad (\text{B.27})$$

Since  $t_{off2} = t_{on} + t_T$ , where  $t_{on}$  is a constant, and,

$$X_T = \begin{bmatrix} 0 \\ \dot{y}_T \end{bmatrix} \quad (\text{B.28})$$

we can solve for the first row of Equation B.24 by pre-multiplying Equation B.27 by  $[1 \ 0]$ :

$$\begin{bmatrix} 1 & 0 \end{bmatrix} \frac{\partial X_T}{\partial X_D} = \begin{bmatrix} 1 & 0 \end{bmatrix} \left[ A(X_T - X_{eoff}) \frac{\partial t_T}{\partial X_D} + e^{At_{off2}} \right] = 0 \quad (\text{B.29})$$

and we can solve for the derivative of  $t_T$  with respect to the state  $X_N$ :

$$\frac{\partial t_T}{\partial X_D} = - \left[ \begin{bmatrix} 1 & 0 \end{bmatrix} A(X_T - X_{eoff}) \right]^{-1} \begin{bmatrix} 1 & 0 \end{bmatrix} e^{At_{off2}} \quad (\text{B.30})$$

Expanding  $A$  and using Equation B.28, we find that:

$$\frac{\partial t_T}{\partial X_D} = -(1/\dot{y}_T) \begin{bmatrix} 1 & 0 \end{bmatrix} e^{At_{off2}} \quad (\text{B.31})$$

To find the second row of Equation 24, we premultiply Equation B.27 by  $[0 \ 1]$ , resulting in:

$$\begin{bmatrix} 0 & 1 \end{bmatrix} \frac{\partial X_T}{\partial X_D} = \begin{bmatrix} 0 & 1 \end{bmatrix} \left[ A(X_T - X_{eoff}) \frac{\partial t_T}{\partial X_D} + e^{At_{off2}} \right] = \frac{\partial \dot{y}_T}{\partial X_D} \quad (\text{B.32})$$

Thus, we can write the derivative for the second GROUND\_OFF mode in the "Short Thrust" case as:

$$\frac{\partial \tilde{X}_T}{\partial \tilde{X}_D} = \left[ \begin{array}{c} -(1/\dot{y}_T) \begin{bmatrix} 1 & 0 \end{bmatrix} \\ \begin{bmatrix} 0 & 1 \end{bmatrix} \left[ A(X_T - X_{eoff}) (-1/\dot{y}_T) \begin{bmatrix} 1 & 0 \end{bmatrix} + I \right] \end{array} \right] e^{At_{off2}} \quad (\text{B.33})$$

and simplify by expanding  $A$ , and substituting for  $X_T$  and  $X_{eoff}$ :

$$\frac{\partial \tilde{X}_T}{\partial \tilde{X}_D} = (-1/\dot{y}_T) \left[ \begin{array}{cc} 1 & 0 \\ (-1 - 2\zeta\omega\dot{y}_T) & -\dot{y}_T \end{array} \right] e^{At_{off2}} \quad (\text{B.34})$$

Having found the individual Jacobians for each of the modes in the vertical hopper, we can compose the overall Jacobian by forming the product of the individual Jacobians according to each of the two mode sequences of interest.

# Bibliography

- Abbas, J.J. and Full, R.J., "Neuromechanical Interaction in Cyclic Movements" in *Biomechanics and Neural Control of Movement*, Ed. by J.M. Winters and P.E. Crago, Springer-Verlag, pp. 177-191, 2000.
- Bailey, S. A., Cham, J. G., Cutkosky, M. R., and Full, R. J., "Biomimetic Mechanisms via Shape Deposition Manufacturing," in *Robotics Research: the 9th International Symposium*, J. Hollerbach and D. Koditschek (Eds), Springer-Verlag, London, 1999.
- Bares, John E. and Wettergreen, D. S. "Dante II: technical description, results, and lessons learned" *International Journal of Robotics Research* v 18 n 7 p 621-649, 1999.
- Berkemeier, M. D. and Desai, K.V., "A Comparison of Three Approaches for the Control of Hopping Height in Legged Robots." Submitted to the *International Journal of Robotics Research*, 1998.
- Brown, I.E. and Loeb, G.E., "A reductionist approach to creating and using neuromusculoskeletal models." In *Biomechanics and Neural Control of Posture and Movement*. (Eds. Winters, J. M. and Crago. P. E.), 1999.
- Cham, J. G., Bailey, S. A., and Cutkosky, M. R., "Robust Dynamic Locomotion Through Feedforward-Preflex Interaction," *ASME IMECE Proceedings*, Orlando, Florida, November 5-10, 2000
- Cham, J. G., Bailey, S. A., Clark, J. E., Full, R. J. and Cutkosky, M. R., "Fast and Robust: Hexapedal Robots via Shape Deposition Manufacturing," To appear in *International Journal of Robotics Research*, 2002.
- Cham, J. G., Stafford, B. and Cutkosky, M. R., "See Labs Run: A Design-oriented Laboratory for Teaching Dynamic Systems," *American Society of Mechanical Engineers, IMECE 2001*, New York, NY, Nov. 11-16, 2001.
- DeCarlo, R., Branicky, M., Pettersson, S., and Lennartson B., "Perspectives and Results on the Stability and Stabilizability of Hybrid Systems" *Proceedings of the IEEE*, July 2000, pp. 1069-1082.

- Franklin, G. F., Powell, J. D. and Workman, M. L., "Digital Control of Dynamic Systems," Addison-Wesley, Menlo Park, 1998.
- Full, R. J., Autumn, K., Chung, J. I., and Ahn, A., "Rapid negotiation of rough terrain by the death-head cockroach," *American Zoologist*, 38:81A, 1998.
- Full, R.J. and Koditschek, D. E., "Templates and Anchors - Neuromechanical hypotheses of legged locomotion on land." In *Designs for Life. The Journal of Experimental Biology*. In press.
- Gabrielli, G., and von Karman, T. H., "What price speed?" *Mechanical Engineering*, vol. 72, no. 10, 1950.
- Garcia, M., Kuo, A., Peattie, A. M., Wang, P. C. and Full, R. J., "Damping and Size: Insights and Biological Inspiration" in *Proc. of the Intl. Symp. on Adaptive Motion of Animals and Machines*, Montreal, Canada, August 2000.
- Hogan, N. Impedance Control: An Approach to Manipulation: Part I-Theory. *Journal of Dynamic Syst., Measure, and Cont.*, 107(1):1-7, 1985a.
- Koditschek, D. E. and Buhler, M., "Analysis of a Simplified Hopping Robot," *International Journal of Robotics Research*, 10:6, 587-605. 1991.
- Komsuoglu, H. and Koditschek, D. E., "Preliminary Analysis of a Biologically Inspired 1-DOF 'Clock' Stabilized Hopper", *Proc. of World Multiconference on Systemics, Cybernetics and Informatics (SCI2000)*, Orlando, USA, July 23-26, 2000, Vol IX, pp 670-675.
- Kuo, A. D. (1999) Stabilization of lateral motion in passive dynamic walking, *International Journal of Robotics Research*, 18(9): 917-930.
- Leijen, D., "Division and Modulus for Computer Scientists," <http://www.cs.uu.nl/~daan/papers/divmodnote-letter.pdf>, 2001.
- McGeer, T., "Passive dynamic walking," *International Journal of Robotics Research*, Vol. 9, No., 2, pp. 62-82, 1990.
- M'Closkey, R.T. and Burdick, J.W., "On the Periodic Motions of a Hopping Robot with Vertical and Forward Motion," *International Journal of Robotics Research*, vol. 12, no. 3, June, 1993, pp. 197-218.

- McMahon, T. A., "Muscles, Reflexes and Locomotion" University Press, Princeton NJ, 1984.
- Merz, R., Prinz, F.B., Ramaswami, K., Terk, M. and Weiss, L., "Shape Deposition Manufacturing," Proceedings of the Solid Freeform Fabrication Symposium, University of Texas at Austin, August 8-10, 1994.
- Mombaur, K. D., "Stability Optimization of Open-loop Controlled Walking Robots," PhD Thesis, Universität Heidelberg, 2001.
- Nanua, P. and Waldron, K. J., "Instability and Chaos in Quadruped Gallop," Trans. ASME Journal of Mechanical Design, Vol. 116, No. 4, 1994, pp1096-1101.
- Ogata, K., "Discrete-Time Control Systems," Prentice-Hall, New Jersey, 1994.
- Orlovsky, G. N., Deliagnia, T. G. and Grilner, S., "Neuronal Control of Locomotion," Oxford University Press, New York, 1999.
- Raibert, M. H., "Legged robots that balance." MIT Press, Cambridge, MA, 1986.
- Ringrose, R., "Self-stabilizing running," PhD Thesis, Massachusetts Institute of Technology, 1997.
- Rossignol, S., Lund, J. P. and Drew, T., "The Role of Sensory Inputs in Regulating Patterns of Rhythmical Movements in Higher Vertebrates," in "Neural Control of Rhythmic Movements in Vertebrates" edited by Cohen, A. H., Rossignol, S. and Grilner S., 1988.
- Saranli, U., Buehler, M. and Koditschek, D.E., "RHex: A Simple and Highly Mobile Hexapod Robot", in The International Journal of Robotics Research, 20(7):616-631, July 2001
- Sastry, S. "Nonlinear Systems: Analysis, Stability and Control". Springer Verlag. 1999.
- Vakakis, A. F., Burdick, J. W., and Caughey, T. K., "An Interesting Strange Attractor in the Dynamics of a Hopping Robot," International Journal of Robotics Research, 10:606-618.
- van Soest, A. J. and Bobbert, M. F., "The contribution of muscle properties in the control of explosive movements" in Biol. Cybernetics 69: 195-204, 1993

Vogel, S. "Better bent than broken" Discover, p62 - 65, May 1995.

Waldron, K. J., "Force motion management in legged locomotion." IEEE Trans on Robotics and Automation. 2 (4): 214-220, 1986.

Weingarten, J. D., Buehler, M., Groff, R. E. and Koditschek, D. E., "Gait Generation and Optimization for Legged Robots," Submitted to the 2003 International Conference on Robotics and Automation.