

# Toward a Formal Framework for Open-Loop Stabilization of Rhythmic Tasks

by

**Haldun Komsuoğlu**

A dissertation submitted in partial fulfillment  
of the requirements for the degree of  
Doctor of Philosophy  
(Electrical Engineering: Systems)  
in The University of Michigan  
2004

Doctoral Committee:

Professor Daniel E. Koditschek, Chair

Professor Jessy W. Grizzle

Professor Arthur D. Kuo

Professor Martin Buehler, McGill University, Canada

Professor Philip J. Holmes, Princeton University

© Haldun Komsuođlu 2004  
All Rights Reserved

For all those who value originality

## ACKNOWLEDGEMENTS

My time in University of Michigan has been enormously rich and enjoyable. I will always be appreciative for all the opportunities that has been given to me. I strongly believe that the seven years I have spent here at Ann Arbor has shaped my personality in significant ways.

There are a number of people that I will always be in debt. At the very top of the list is my advisor, Daniel E. Koditschek. Over the years I have learned a great deal from Dan. Most importantly, he taught me how to express myself — an ability I horribly lacked when I first met him. Besides his invaluable technical guidance Dan has helped me in several occasions when my timid personality has interfered with my work and life. I thank him for all his understanding, for his honesty and patience, for our inspiring conversations and for above all giving me an opportunity that was only in my wildest dreams.

I thank Martin Buehler for conceiving RHex which constituted the basis of my work during my doctoral studies. His unmatched intuition on how physical machines work will always be a mystery for me. I thank Alfred Rizzi for his assistance and input during the implementation of RHex and RiSE.

Another significant part of my doctoral adventure has been my lab-mates — an incredibly resourceful and intelligent bunch. I thank Noah Cowan, my first mentor, for introducing me to the laboratory culture. I thank Eric Klavins for his guidance during our short but exciting work. I will always be jealous of his creativity. I thank

Uluc Saranli — the first Turk of the lab — for his technical abilities and building RHex, but more importantly for introducing me to the ballroom dance team. I thank Richard Groff for many stimulating conversations and his unending stream of knowledge. In fact, some of the results in this thesis can be attributed to an informal two minute chat we had four years ago. Another indispensable member of our lab is Pei-Chun Lin. I thank Pei-Chun for all his help in a number of projects and particularly showing me how to do mechanical design and implementation. I thank Richard Altendorfer, Joel Weingarden, Gabriel Lopes, Eric Westerwelt for all their assistance and friendship over the years.

I thank the staff of the Advanced Technology Laboratory and the EECS department here at the University of Michigan, particularly Karen Alexa, Karen Coulter, Kelly Corimer and Wendy Gillikin, and Linda Cox, without whom I could not get anything done.

The life in Ann Arbor would be meaningless without my friends. In general, it is very hard to come by good friends but I must admit I have been spoiled. I thank Melih Cakmakci and Tolga Senoguz for numerous Turkish nights and parties. I thank the University of Michigan Ballroom Dance Team for accepting me and letting me be a part of an exceptionally wonderful organization. I thank Sibylle Thies, my three year roommate and a member of the dance team, for being patient with me when living with me was not so easy. I specially thank Jia Lu, my one and only dance partner. I apologize her for all the meaningless and childish fights we had before each and every competition and thank her for being such a wonderful partner.

I thank my parents for providing me a good education and an unconstrained and comfortable life in general. Their unconditional love and trust has always been an implicit driving force for me even at times when I lost all self-confidence. Special thanks goes to my grandmother for teaching me how to cook turkish pies.

Finally, I thank my girlfriend, Malgorzata Ochocinska, for always being there for me even when I could not stand myself. She is by far the most lovable person I know of and I feel lucky to be with her. By the power of cuteness...

## PREFACE

The wheel has been the primary means of terrestrial transportation since its invention in 4000 B.C. The popularity of wheeled vehicles can be attributed to the ease of their control, which requires no or very little sensory feedback in most scenarios. Unfortunately, the operational domain of wheeled vehicles is severely limited in natural settings where the conditions favorable to wheeled locomotion are seldom met. On the other hand, legged animals exhibit extraordinary locomotion performance over highly unstructured and unstable surfaces that no wheeled vehicle can even approach. Legs are also highly versatile tools that can serve for purposes other than locomotion, such as manipulation of external objects. However, in the field of robotics, these desirable features of legged systems have been overshadowed by the difficulty of construction and control of legged platforms which remains a major obstacle to producing physically viable legged systems. Combining the ease of control of wheeled approaches with the performance of legs may likely lead to a quantum leap in our ability to move in terrestrial settings.

Recent collaborations between engineers and biologists have led to the identification of a family of biologically-inspired control principles for legged locomotion. It was shown that the passive mechanical musculoskeletal system plays a crucial part in the control of legged locomotion. In fact, in some extreme cases the active control takes the form of a purely feed-forward excitation of this passive mechanical system. RHex — a highly dexterous autonomous hexapod robot — has demonstrated that task-level

open-loop control can in fact give rise to exceptional performance over rough surfaces and may be the key to the eventual development of real world products.

From an engineering point of view, open-loop controllers are very desirable solutions due to their significantly simpler structure which does not require sensors or related infrastructure. In essence, the ability to control legged locomotion by feed-forward controllers brings ease of control — a feature typically attributed to wheeled vehicles — into the field of legged robotics. Unfortunately, to this date the design of open-loop controllers is still a “black art” in which the intuition of the researcher acts as the only design tool.

This thesis marks the beginning of a formal framework to design and verify open-loop controllers for dynamical legged locomotion. To this end we consider a very simple open-loop controlled dynamical model — a clock driven 1-DOF hopper — to investigate the basic principles of open-loop control with the overarching goal of identifying sufficient (and hopefully necessary) conditions for stable locomotion. We present an analysis which allows us to study arbitrary excitation patterns. We discuss a computational algorithm for the design of open-loop controllers based on these analytic results. Furthermore, we introduce a family of hierarchical stride-to-stride adaptation laws that take advantage of this open-loop setup. Our numerical studies suggest that some key ideas discovered in this simple illustrative model indeed extend to a wide family of physically relevant setups.

# TABLE OF CONTENTS

<b>DEDICATION</b> . . . . .	ii
<b>ACKNOWLEDGEMENTS</b> . . . . .	iii
<b>PREFACE</b> . . . . .	vi
<b>LIST OF TABLES</b> . . . . .	x
<b>LIST OF FIGURES</b> . . . . .	xi
<b>LIST OF APPENDICES</b> . . . . .	xx
<b>NOTATION</b> . . . . .	xxi
<b>CHAPTERS</b>	
1 Introduction . . . . .	1
1.1 Motivation and Scope . . . . .	1
1.2 Biological Inspiration . . . . .	4
1.2.1 Terrestrial Locomotion . . . . .	4
1.2.2 Functional Biomimesis . . . . .	6
1.3 Control of Legged Locomotion . . . . .	10
1.3.1 Complexity must be Reduced . . . . .	10
1.3.2 Feedback Control . . . . .	11
1.3.3 Open-Loop Control . . . . .	15
1.3.4 Adaptation of Open-Loop Control . . . . .	18
1.4 Contributions of This Thesis . . . . .	21
1.4.1 A Novel Clock Driven 1-DOF Hopper Model . . . . .	21
1.4.2 A Family of Adaptation Mechanisms . . . . .	21
1.4.3 Extensions of the Basic Results . . . . .	22
2 A Clock Driven 1-DOF Hopper . . . . .	24
2.1 The Model . . . . .	24
2.1.1 A Mechanical Hopper . . . . .	24

2.1.2	A Clock Controller . . . . .	28
2.1.3	A Unidirectional Coupling Scheme . . . . .	30
2.2	Groundwork for the Analysis . . . . .	31
2.2.1	Preferred Coordinates and Related Transformations . . . . .	31
2.2.2	Basic Setup of the Poincaré Analysis . . . . .	35
2.3	Mode Sequences . . . . .	36
2.3.1	Definition and Significance . . . . .	36
2.3.2	Classes of Mode Sequences . . . . .	37
2.4	Fundamental Repeatable Mode Sequences . . . . .	40
2.4.1	Return Map Derivation . . . . .	41
2.4.2	Mode Sequence Domain Derivation . . . . .	47
2.4.3	Derivation of Fixed Points . . . . .	49
2.4.4	Local Stability Analysis . . . . .	53
2.5	Numerical Studies . . . . .	71
2.5.1	Typical Stable Behaviors of the Coupled System . . . . .	71
2.5.2	Bifurcation Studies . . . . .	72
2.5.3	Numerical Assessment of the Stability Properties . . . . .	79
2.6	Extensions of the Analysis . . . . .	84
2.6.1	Open-Loop Controller Design . . . . .	84
2.6.2	The Role of Mechanical Properties . . . . .	85
3	Feedback Adjustment of the Clock Driven 1-DOF Hopper . . . . .	88
3.1	Feedback Controller . . . . .	90
3.1.1	A Family of Tuning Laws . . . . .	92
3.1.2	A Tuning Procedure . . . . .	92
3.2	The Closed-Loop Setting . . . . .	93
3.2.1	Derivation of the Return Map . . . . .	93
3.2.2	Local Stability Analysis . . . . .	95
3.3	Improving the Rate of Convergence . . . . .	100
3.3.1	Motivation . . . . .	100
3.3.2	A Family of Synchronizer Tuning Laws . . . . .	101
3.3.3	A Numerical Demonstration: Linear Synchronizer . . . . .	103
3.4	Enlarging the Basin of Attraction . . . . .	105
3.4.1	Motivation . . . . .	105
3.4.2	A Sequential Composition Procedure . . . . .	107
3.5	A Hybrid Tuning Law . . . . .	115
4	Conclusion . . . . .	118
	<b>APPENDICES . . . . .</b>	<b>123</b>
	<b>BIBLIOGRAPHY . . . . .</b>	<b>196</b>

## LIST OF TABLES

### Table

2.1	The convergence speed measured for the runs presented in Figure 2.19.	81
C.1	The sign of partial derivatives are related. Table summarizes those combination of sings that occur for softening and hardening springs. The linear spring is not included in this table since it is guarantied to be unstable. The last column indicates the limit characteristic of a fixed point if it lives in the corresponding partition of the Poincaré space, $\mathcal{P}$ .	192

# LIST OF FIGURES

**Figure**

2.1	Mechanical hopper (a) in stance phase; (b) in aerial phase; (c) at ground collision. . . . .	25
2.2	The top sketch depicts a typical state trajectory of the mechanical hopper, $\tilde{\mathbf{f}}^t(\mathbf{x}_0, \xi)$ , with constant stiffness, $\xi$ , where the four cells in the partition of the physical state space, $\mathcal{X}$ , are listed in the accompanying legend. Plastic collision at touchdown, $\mathbf{x}_{TD}$ , results in a reduction in total mechanical energy defined by the plastic collision map, $\tilde{\mathbf{c}}(\mathbf{x}_{TD})$ . The bottom plot is the hybrid potential of the mechanical hopper, $U_\xi(x_1)$ , composed of gravitational and Hook's law spring potentials. .	27
2.3	An illustration of a clock controller. It consists of a clock and an output function, $\psi(\theta)$ , which is an $N$ -cell piecewise constant function mapping the clock phase to the scalar output. The output function, $\psi(\theta)$ , is parameterized by the shape configuration, $\mathbf{o} \in \mathcal{O}$ . The controller output, $\psi(\theta(t))$ , is a piece-wise constant periodic signal. . . . .	29
2.4	Clock driven 1-DOF hopper setting. The mechanical hopper and the clock controller are forward coupled such that the periodic output of the controller, $\psi(\theta(t))$ , modulates the leg stiffness, $\xi$ . The clock controller is parameterized by the clock period, $T_c$ , and the shape configuration, $\mathbf{o}$ , which are considered adjustable. . . . .	30
2.5	Coordinate systems and the transformations that relate them. . . . .	33
2.6	A numerically computed partition of the Poincaré state space, $\mathcal{P}$ , according to the common mode sequences, $\sigma$ , for a coupled system with 2-cell output function whose parameters are given in the title. Some of the mode domains are pointed out along with the set of Poincaré states that lead to ground crash. . . . .	38
2.7	A typical output function with $N$ -cells (left) and a typical hopper state trajectory in the physical coordinate system, $\mathcal{X}$ , between two consecutive Poincaré samples (right). Crucial events along the trajectory are indicated by markers whose list can be found in the legend below. . .	42

2.8	The trace, $\text{tr}$ , (left) and the discriminant, $\Delta$ , (right) functions and their respective bounds plotted against the clock period, $T_c$ , for a given fixed point, $\mathbf{p}^*$ , in $\mathcal{P}_-$ . The physically uninteresting interval of the clock period axis is shaded. . . . .	56
2.9	Eigenvalues, $\lambda_{\pm}$ , of the Jacobian evaluated at a fixed point, $\mathbf{p}^*$ in $\mathcal{P}_-$ as a function of the clock period, $T_c$ , over the admissible clock period values, $T_c \in (T_c^0(\alpha^*), \infty)$ . Circles indicate the location of the eigenvalues at $T_c = T_c^0(\alpha^*)$ . . . . .	58
2.10	The trace, $\text{tr}$ , (left) and the discriminant, $\Delta$ , (right) functions and their respective bounds plotted against the clock period, $T_c$ , for a given fixed point, $\mathbf{p}^*$ , in $\mathcal{P}_+$ . The physically uninteresting interval of the clock period axis is shaded. . . . .	62
2.11	The eigenvalues of the Jacobian, $\mathbf{J}(\mathbf{p}^*)$ , evaluated at a fixed point, $\mathbf{p}^*$ , in $\mathcal{P}_+$ . The locus is parametrized by the magnitude of $\varphi_e'(\alpha^*)$ which determines whether the eigenvalues are (a) inside or (b) outside the unit circle, determining, in turn the stability of the fixed point, $\mathbf{p}^*$ . . . . .	64
2.12	Typical evolution of the body height in a neutrally stable fundamental repeatable mode sequence, $\mathcal{GAG}(6, 6)$ . The mechanical system is lossless, $\mu = 0$ , and the shape function, $\psi(\theta)$ , takes a 6-cell triangular form. The top plot shows the body height across the entire experiment span. The bottom plots zoom in to the very last two cycles of the run. The bottom-left plot shows the body height. The circle markers indicate the event of clock reset and the square markers point out where triangle portion of the excitation shape concludes. Shape function, $\psi(t)$ , is depicted in the bottom-left plot. . . . .	73
2.13	Numerically computed sets of stable shape configurations, $\mathcal{O}_s(\mu)$ , for a coupled system driven by a 2-cell shape function, $\psi(\theta)$ , operating in $\mathcal{GAG}(2, 2)$ . The shape function is parameterized by the first cell duration, $\beta_1$ , and the difference between the stiffness in the first and second cells, $\Delta\xi := \xi_1 - \xi_2$ . The set of stable shape configurations, $\mathcal{O}_s(\mu)$ , is a function of the damping coefficient, $\mu$ . Two different damping levels depict the relationship between mechanical losses and actuation. . . . .	74
2.14	Typical evolution of the body height in a stable fundamental repeatable mode sequence, $\mathcal{GAG}(4, 5)$ . In these experiments the mechanical system is lossy and the shape function, $\psi(\theta)$ , takes a 4-cell piece-wise constant approximation of a triangle form. The top plots show the body height across the entire experiment span. The two plots in the bottom zoom in to the very last two cycles of the run. The bottom-left plot shows the limit mechanical behavior. The circle markers indicate the event of clock reset, $\theta = 0$ , and the square markers point out where triangle portion of the excitation shape concludes, $\theta = \theta_4$ . The bottom-right plot depicts the shape function, $\psi(t)$ , during the last two periods of clock. . . . .	75

- 2.15 Typical evolution of the body height in a stable fundamental repeatable mode sequence,  $\mathcal{GAG}(40, 41)$ . In these experiments the mechanical system is lossy and the shape function,  $\psi(\theta)$ , takes a 41-cell piece-wise constant approximation of a triangle form. The top plots show the body height across the entire experiment span. The two plots in the bottom zoom in to the very last two periods of the run. The bottom-left plot shows the limit mechanical behavior. The circle markers indicate the event of clock reset,  $\theta = 0$ , and the square markers point out where triangle portion of the excitation shape concludes,  $\theta = \theta_{40}$ . The bottom-right plot depicts the shape function,  $\psi(t)$ , during the last two clock periods. . . . . 76
- 2.16 Numerically computed locus of eigenvalues as the clock period,  $T_c$ , varies. The top plot depicts the forcing function,  $\rho(\alpha)$ , at the chosen shape configuration,  $\mathbf{o}$ , whose two unity crossings in the delay interval,  $\mathcal{D}$ , define two fixed points,  $\mathbf{p}_-^* \in \mathcal{P}_-^*$ , and  $\mathbf{p}_+^* \in \mathcal{P}_+^*$ . The middle plots show the loci of the two eigenvalues,  $\lambda_-$  (blue) and  $\lambda_+$  (red), for each fixed point: left is for  $\mathbf{p}_+^*$ ; right is for  $\mathbf{p}_-^*$ . The asterisks indicate the eigenvalues at the smallest admissible clock period,  $T_c = T_c^0$ . The bottom plots depict the magnitudes of the eigenvalues,  $\lambda_{\pm}$ , as a function of the clock period,  $T_c$ . . . . . 77
- 2.17 Numerically computed locus of eigenvalues as the damping coefficient,  $\mu$ , varies. The shape configuration,  $\mathbf{o}$ , and the clock period,  $T_c$ , are kept constant. The damping coefficient spans an interval,  $(\mu_-, \mu_+)$ , which contains the lossless case,  $\mu = 0$ . The top plot shows the forcing function,  $\rho(\alpha)$ , for gainy,  $\mu = \mu_-$ , lossless,  $\mu = 0$ , and lossy,  $\mu = \mu_+$ , cases. We consider only those damping values that give rise to fixed points,  $\mathbf{p}_{\pm}^*$ . The middle plots show the loci of the two eigenvalues,  $\lambda_-$  (blue) and  $\lambda_+$  (red), for each fixed point: left is for  $\mathbf{p}_+^*$ ; right is for  $\mathbf{p}_-^*$ . The asterisks indicate the eigenvalues at  $\mu = \mu_-$ . The bottom plots depict the magnitudes of the eigenvalues,  $\lambda_{pm}$ , as a function of the damping,  $\mu$ . . . . . 78
- 2.18 The discrete flow of the Poincaré states for three different mechanical dissipation settings: lossy,  $\mu > 0$ ; lossless,  $\mu = 0$ ; and gainy,  $\mu < 0$ . The boundaries of the valid mode domain,  $\mathcal{P}(\sigma)$ , are indicated in each plot. The (blue) star indicates the (only) valid fixed point. The initial and final conditions of each flow are marked by circles and squares, respectively. The lossy setting is locally asymptotically stable. The lossless case demonstrates quasi-periodic neutrally stable behavior. Finally, the gainy setup is unstable. . . . . 80

2.19	The discrete Poincaré map at eight different mechanical dissipation levels: (a) $\mu = 0$ ; (b) $\mu = 0.01$ ; (c) $\mu = 0.02$ ; (d) $\mu = 0.03$ ; (e) $\mu = 0.04$ ; (f) $\mu = 0.05$ ; (g) $\mu = 0.06$ ; and (h) $\mu = 0.07$ . In each plot the valid mode sequence domain, $\mathcal{P}(\sigma)$ , is indicated. The initial condition of each experiment is displaced from its stable fixed point by a constant distance. The return map is iterated 10 times. The convergence speed is measure by the surrogate in (2.51). The convergence speed for each run can be found in Table 2.1. . . . .	82
2.20	The basin of the stable fixed point, $\mathbf{p}_+^*$ , for six different damping levels: (a) $\mu = 0$ ; (b) $\mu = 0.002$ ; (c) $\mu = 0.004$ ; (d) $\mu = 0.01$ ; (e) $\mu = 0.03$ ; and (f) $\mu = 0.07$ . In all plots except the lossless case in (a) the color coded area within the valid cell, $\mathcal{P}(\sigma)$ , is the numerically computed domain of attraction of the stable fixed point which is indicated by a (blue) star. In the lossless case in (a) the color coded area is the invariant set under the return map, $\mathbf{r}(\mathbf{p})$ . However, the discrete map does not converge to the fixed point, $\mathbf{p}^*$ . . . . .	83
2.21	The basin of the stable fixed point, $\mathbf{p}_+^*$ , for six different clock periods: (a) $T_c = 0.9$ ; (b) $T_c = 2.0$ ; (c) $T_c = 3.0$ ; (d) $T_c = 4.0$ ; (e) $T_c = 5.0$ ; and (f) $T_c = 6.0$ . The color coded area within the valid cell, $\mathcal{P}(\sigma)$ , is the numerically computed domain of attraction of the stable fixed point which is indicated by a (blue) star. . . . .	84
3.1	The generic structure of the closed loop system, consisting of a clock controller and a tuning mechanism. The states of the coupled system, $(\nu, \alpha)$ , are sampled at the event of clock reset, $\theta = 0$ . The tuning mechanism updates the clock parameters, the clock period, $T_c$ , and the shape configuration, $\mathbf{o}$ , as described by a tuning law — a functional relationship between the Poincaré states, $(\nu, \alpha)$ , and the clock period, $T_c$ . The adjustments to the shape, $\mathbf{o}$ , are performed such that the durations of the stance cells, $\beta_i$ , $i = 1, \dots, e, N$ , are kept constant. (Note that $e + 1 = N$ is not necessarily true.) . . . . .	91
3.2	A conceptual comparison between the locus of the eigenvalues of open-loop (left) and synchronizer closed-loop (right) systems for a fixed point, $\mathbf{p}^*$ , in $\mathcal{P}_+^*$ class. A tuning law, $\tau(\nu, \alpha)$ , which satisfies the conditions in Proposition 6 results in closed-loop eigenvalues that are closer to the origin that the eigenvalues of the underlying open-loop system, as depicted in the sketch on the right. . . . .	99
3.3	The valid domain, $\mathcal{P}(\sigma)$ , of the closed-loop system employing the linear synchronizer tuning law in (3.15) at different synchronization gains, $K_s$ . Note that the valid delay interval, $\mathcal{D}(\sigma)$ , remains unchanged, whereas the bounds on the speed state, $\nu$ , gets shifted as a result of the changing synchronization gain, $K_s$ . . . . .	102

3.4	Convergence in a closed-loop system that uses the tuning law in (3.15). The effect of different synchronization gains, $K_s$ , is demonstrated. Each subplot contains: body height (top); instantaneous clock period, $T_c$ (middle); and the Poincaré sequence, $\mathbf{p}_k$ (bottom). The bounded area in the Poincaré space is the valid domain, $\mathcal{P}(\sigma)$ , of the closed-loop return map, $\mathbf{r}(\mathbf{p})$ . . . . .	105
3.5	The domain of attraction of a closed-loop system that uses the tuning law in (3.15). The subplots exhibit the basins (solid filled) and the valid domain (bounded area) at different settings of the synchronization gain, $K_s$ . . . . .	106
3.6	A conceptual illustration of the dependence between the clock period, $T_c$ , and the location of the valid domain, $\mathcal{P}(\sigma, T_c)$ , and the fixed point, $\mathbf{p}^*(T_c)$ . . . . .	108
3.7	An illustration of the naming convention employed in the composition discussion. Note how the changes in the clock period lead to translations in the valid domain, $\mathcal{P}_k(\sigma)$ , fixed point, $\mathbf{p}_k^*$ , and the basin, $\mathcal{P}_k^*$ . The target related expressions are identified by the lack of subscripts. . . . .	109
3.8	An illustration of the relative positions of the boundaries of the $k^{\text{th}}$ valid domain, $\mathcal{P}_k(\sigma)$ , the $k^{\text{th}}$ fixed point, $\mathbf{p}_k^*$ , and the $k^{\text{th}}$ Poincaré state, $\mathbf{p}_k$ . This sketch shows a case where the target fixed point speed, $\nu^*$ , is smaller than the $k^{\text{th}}$ cycle speed, $\nu$ . . . . .	112
3.9	A typical run of the close-loop system employing the composition tuning law in (3.18). The figure contains three plots: the body height of the mechanical hopper, $x_1$ , (top); the instantaneous clock period, $T_c$ , as it is dictated by the composition tuning law (middle); and the the discrete flow of the Poincaré states, $\mathbf{p}_k$ . The bottom plot indicates the valid domain of the open-loop system, $\mathcal{P}(\sigma)$ . . . . .	113
3.10	A list of snapshots during a typical run of a closed-loop system that employs the composition tuning law in (3.18). The far left column indicates the cycle index. Each plot represents the state of the closed-loop system at a particular cycle and contains the valid domain of the open-loop system, $\mathcal{P}(\sigma)$ ; target fixed point, $\mathbf{p}^*$ ; cycle valid domain, $\mathcal{P}_k(\alpha)$ ; the cycle fixed point, $\mathbf{p}_k^*$ ; and the cycle state, $\mathbf{p}_k$ . . . . .	114
3.11	The hierarchical structure of the hybrid tuning law in (3.22). The synchronizer input, the nominal period, $T_n$ , replaces the the original period input of the clock controller and offers the same parameterization of the valid domain, $\mathcal{P}(\sigma)$ , and fixed point, $\mathbf{p}^*$ . . . . .	116

3.12	The Poincaré state flows of the closed-loop system that employs the hybrid tuning law in (3.22). Two different gain settings are exhibited. In the top figure the synchronizer gain, $K_s = 0.1$ , is small, and therefore, the composition gain has to be set to a small value, $K_c = 0.05$ . In the bottom plot a larger synchronization gain, $K_s = 0.5$ , allows for a higher composition gain, $K_c = 0.1$ . Recall composition gain, $K_c$ , determines the speed of convergence. . . . .	117
B.1	The user specifies the desired behavior by the apex height of hopping, $x_1^{apex}$ . The open-loop controller design is a two step process: 1) design of a “proper” shape function, $\psi(\theta)$ ; and 2) computation of the clock period, $T_c$ . . . . .	129
B.2	The design of the shape function, $\psi(\theta)$ , starts with selection of a nominal stiffness, $\xi_n$ . Next, those entries of the shape configuration, $\mathbf{o}$ , are computed such that the forcing function, $\rho(\alpha)$ , intersects with the unity where its first derivative falls within the bounded interval in (B.5). Finally, the durations of the lift-off cell, $\beta_e$ , and the last cell, $\beta_N$ , are determined such that the invariant delay space, $\mathcal{D}^*$ , captures this unity crossing point. . . . .	132
B.3	Illustration of typical action term plots (top) and associated discontinuous change in the shape function, $out(\theta)$ (bottom). The left side plots depict a relaxation action and the right side plots correspond to a stiffening actions. . . . .	138
B.4	A typical mechanical state trajectory during a single action operating regime between two consecutive Poincaré samples (right) and a typical single action shape function, $\psi(\theta)$ (left). Crucial events along the trajectory are indicated by markers whose list can be found in the legend below. . . . .	141
B.5	The single action forcing function, $\rho(\alpha)$ , and its bounds are depicted on the bottom sketch. The plots on the top illustrate the typical form of the single (stiffening) action, $l_1(\alpha)$ , (left) and the single action loss term, $\epsilon(\alpha)$ (right). . . . .	143
B.6	Demonstration of the closeness between the approximate and original expressions of the forcing function and the remaining time function in an undesirable setting. Top shows a 4-cell triangular shape function, $\psi(\theta)$ , for which the plots are generated. Middle two plots compare the original forcing function, $\rho(\alpha)$ , to the approximate forcing function, $\hat{\rho}(\alpha)$ , and the basic forcing function, $\hat{\hat{\rho}}(\alpha)$ , along with the analytic bounds. The bottom plot compares the remaining time function, $\varphi_e(\alpha)$ , to its approximation, $\hat{\varphi}_e(\alpha)$ . . . . .	160

B.7	Demonstration of the closeness between the approximate and original expressions of the forcing function and the remaining time function in an typical small action operating regime. Top shows a 40-cell triangular shape function, $\psi(\theta)$ , for which the plots are generated. Middle two plots compare the original forcing function, $\rho(\alpha)$ , to the approximate forcing function, $\hat{\rho}(\alpha)$ , and the basic forcing function, $\hat{\hat{\rho}}(\alpha)$ , along with the analytic bounds. The bottom plot compares the remaining time function, $\varphi_e(\alpha)$ , to its approximation, $\hat{\varphi}_e(\alpha)$ . . . . .	161
B.8	The top plot illustrates numerically evaluated basic forcing error as a function of two properties of the shape function, $\psi$ : its minimum, $\min(\psi)$ ; and its standard deviation, $\text{std}(\psi)$ . The magnitude of the error is color coded such that error grows from blue to red monotonically. The bottom plots demonstrate the shape functions, $\psi(\theta)$ , and the comparison of the original forcing function, $\rho(\alpha)$ , with the basic forcing function, $\hat{\hat{\rho}}(\alpha)$ . The pair on the left corresponds to the worst while the pair on the right depicts the best case observed in this study.	162
B.9	The limit approximation error in the basic forcing function, $\hat{\hat{\rho}}(\alpha)$ , as the number of cells in the shape function, $N$ , increases. The continuous shape template is a ramp function. The small action approximate is obtained by uniform sampling-and-hold. . . . .	163
B.10	The limit approximation error in the basic forcing function, $\hat{\hat{\rho}}(\alpha)$ , as the number of cells in the shape function, $N$ , increases. The continuous shape template is a triangle function. The small action approximate is obtained by uniform sampling-and-hold. . . . .	164
B.11	The limit approximation error in the basic forcing function, $\hat{\hat{\rho}}(\alpha)$ , as the number of cells in the shape function, $N$ , increases. The continuous shape template is a half period of a sinusoid function. The small action approximate is obtained by uniform sampling-and-hold. . . . .	165
B.12	Vector representations of basic actions, $\hat{l}_i$ , for stiffening (top) and relaxation (bottom) actions. On the left column the action vectors, $\mathbf{l}_i$ , are depicted in the action space, $\mathcal{L}$ . The sketches on the right are the corresponding basic action terms, $\hat{l}_i(\hat{\alpha})$ , plotted against the normalized delay, $\hat{\alpha} := 2\gamma_N\alpha$ . The phase of the action vector, $\angle\mathbf{l}_i$ , indicates the delay where the associated basic action term, $\hat{l}_i(\alpha)$ , has its local extrema points. The action space, $\mathcal{L}$ , is partitioned into two cells: 1) a clock-wise half (light gray); and 2) a counter clock-wise half (dark gray). The basic action term, $\hat{l}_i(\alpha)$ , is monotonically decreasing in the former and monotonically increasing in the latter. . . . .	168

B.13	An illustration of the action vectors for a two action fundamental repeatable mode sequence, $\mathcal{GAG}(2, N)$ , as those investigated in [1]. The first action is a stiffening action, $l_1$ , which is followed by a relaxation action, $l_2$ . By definition, the first action vector, $\mathbf{l}_1$ , is aligned with the cos-axis, $S_1 = 0$ . The cumulative action vector, $\mathbf{l}_c$ , (black arrow) is the vector sum of the individual action vectors, $\mathbf{l}_i$ (gray arrows). The phase of the cumulative action vector, $\angle \mathbf{l}_c$ , determines the delay states where the basic cumulative action term, $\hat{L}(\alpha)$ , has local extrema. The range of the basic cumulative action term is a function of the magnitude of the cumulative action vector, $ \mathbf{l}_c $ . Those value at the end of each ray indicates the angle between the ray and the positive cos-axis in the counter clock-wise direction. All other angles are explicitly indicated.	169
B.14	The approximate invariant delay interval. Those numbers at the end of each ray indicate the angle between the ray and the positive cos-axis in the counter clock-wise direction. . . . .	171
B.15	An illustration of the spike shape function. The profile is partitioned into three cells: 1) spike portion; 2) lift-off cell; and 3) nominal cell. The spike portion is composed of $2L - 1$ cells of identical duration where $L$ is the number of levels in output spans in spike portion. The target operating regime is a fundamental repeatable mode sequence. The mechanical system transitions into the aerial mode in the lift-off cell, $\Theta_e$ . The ground contact occurs in the nominal cell, $\Theta_N$ . Notice that $N = e + 1$ . . . . .	173
B.16	The flow chart of the (approximate) spike shape function, $\psi(\theta)$ , design procedure. Each step refers to those equations that govern the computational process to be performed. . . . .	174
C.1	1-DOF Vertical Watch Spring Model (VWSM) in ground contact, $\mathcal{G}$ , (right); and in flight, $\mathcal{A}$ , (left). . . . .	180
C.2	The relationship between the area enclosed by the level set, $\mathcal{M}_\xi^h$ , and the potential energy function, $U_\xi(x_1)$ . . . . .	187
C.3	Illustrations of the three spring types: softening (left); linear (middle); and hardening (right). . . . .	188
C.4	Typical neutrally stable orbits of clock driven lossless cyclic systems for three spring types are illustrated: softening spring; linear spring; and hardening spring. For each case we present two plots. The top plots are the generalized position plotted against the time. The markers indicate where the clock discontinuously changes the spring potential. In the bottom plots we depict the Poincaré map. Notice that the neutrally stable fixed points of the softening and hardening springs are located at different delay values. The orbit of the linear spring is not a closed curve but the delay state monotonically increases. . . . .	191

C.5	Numerically computed energy map, $\Delta H(\mathbf{p})$ , (right) functions for a lossy softening spring, $U_\xi(x_1) = -\frac{2\xi}{3}\text{sign}(x_1)x_1^{1.5}$ , driven by a 2-cell shape function, (left) and phase map, $\Delta\phi(\mathbf{p})$ . Top images show the surface plot with perspective where as the bottom ones are color maps of the same functions for easy identification of the gradient. . . . .	192
C.6	Numerically computed energy map, $\Delta H(\mathbf{p})$ , (right) functions for a lossy hardening spring, $U_\xi(x_1) = -\frac{\xi}{4}x_1^4$ , driven by a 2-cell shape function, (left) and phase map, $\Delta\phi(\mathbf{p})$ . Top images show the surface plot with perspective where as the bottom ones are color maps of the same functions for easy identification of the gradient. . . . .	193
C.7	A conceptual illustration of the two possible flows of the mechanical system at a fixed point of the Poincaré map. In this example we consider a clock controller with a two cell shape function, $\psi(\theta)$ , shown on the right. Our numerical studies suggest that the softening springs give rise to stable limit behaviors of type (a) and hardening springs result in stable limit behaviors of type (b). . . . .	194
C.8	Mode sequence partitions numerically computed over a 80x40 grid on the Poincaré space, $\mathcal{P}$ , for prismatic hoppers with three types of springs: (a) a softening spring, $U_\xi(x_1) = -\frac{2\xi}{3}\text{sign}(x_1)x_1^{1.5}$ ; (b) Hooke's law spring, $U_\xi(x_1) = -\frac{\xi}{2}x_1^2$ ; and (c) a hardening spring, $U_\xi(x_1) = -\frac{\xi}{4}x_1^4$ . The grid elements that are in the $\mathcal{GAG}(1, 2)$ mode sequence partition are marked by (blue) dots to indicate convergence to the fixed point, $\mathbf{p}$ , and by (red) crosses to indicate unstable operation. Hence, region of blue dots approximate the domain of attraction of the fix point in $\mathcal{GAG}(1, 2)$ whose percentage to the entire $\mathcal{GAG}(1, 2)$ partition is (a) %51.31579 for softening spring; (b) %27.01754 for Hooke's law spring; and (c) %9.65517 for hardening spring . . . . .	196

# LIST OF APPENDICES

## APPENDIX

A	Dimensionless Physical Coordinate System . . . . .	124
B	Open-Loop Controller Design . . . . .	127
C	Clock Driven 1-DOF Generalized Mechanical Systems . . . . .	178

# NOTATION

## Generic Definitions

### Scalar Valued Functions

$\pi_i$   $i^{\text{th}}$  entry of a given vector

### Scalar Valued Variables

$s$  A dummy variable

## 1-DOF Hopper Analysis

### Sets and Spaces

$\mathcal{K}$  Set of admissible physical stiffness values

$\mathcal{B}$  Set of admissible physical damping values

$\mathcal{X}$  Physical mechanical hopper state space

$\mathcal{X}_A$  Aerial cell in the partition of  $\mathcal{X}$

$\mathcal{X}_G$  Stance cell in the partition of  $\mathcal{X}$

$\mathcal{X}_G^0$	Ground collision cell in the partition of $\mathcal{X}$
$\mathcal{X}_G^-$	Touchdown cell in the partition of $\mathcal{X}$
$\mathcal{X}_G^+$	Lift-off cell in the partition of $\mathcal{X}$
$\partial\mathcal{X}_G^-$	Touchdown boundary
$\partial\mathcal{X}_G^+$	Lift-Off boundary
$\partial\mathcal{G}^*$	Ground crash boundary
$\Theta$	Clock phase space
$\mathcal{O}$	Clock controller output shape configuration space
$\mathcal{E}_i$	Normalized energy-phase space for $i^{\text{th}}$ cell
$\Sigma$	Poincare section in the preferred coupled state space
$\mathcal{P}$	Poincaré state space
$\mathcal{V}$	Poincaré cycle energy state space
$\mathcal{D}$	Poincaré delay state space
$\mathcal{M}$	Set of realizable mode sequences
$\mathcal{T}$	Admissible clock period interval
$\mathcal{L}$	Action space — a 2-dimensional cartesian space where the action vectors, $\mathbf{l}_i$ , are defined.

### String Valued Variables

$\sigma$	Mode sequence string
----------	----------------------

### Scalar Valued Variables

$x_1$	Physical height of the body
$\theta$	Clock phase
$\eta$	Normalized total mechanical energy
$\phi$	Normalized mechanical phase
$\nu$	Poincare state cycle energy entry
$\alpha$	Poincare state phase difference entry
$\hat{\alpha}$	Normalized Poincare state delay state

### Vector Valued Variables

$\mathbf{x}$	Physical mechanical state vector
$\mathbf{y}$	Normalized mechanical state vector
$\mathbf{e}$	Energy-Phase state vector
$\mathbf{p}$	Poincare map state vector

### Scalar Valued Functions

$U_\xi$	Mechanical potential energy of the hopper parametrized by the adjustable spring stiffness, $\xi$
$H_\xi$	Total mechanical energy of the hopper parametrized by the adjustable spring stiffness, $\xi$
$\psi$	Clock controller output shaping function
$b_i^j$	Phase component of the map from $\mathcal{E}_i$ to $\mathcal{E}_j$
$a_i^j$	Energy coefficient of the map from $\mathcal{E}_i$ to $\mathcal{E}_j$

$H_i^j$	Affine translation in the map from $\mathcal{E}_i$ to $\mathcal{E}_j$
$P$	Affine phase transformation in $\mathbf{n}^{-1}$
$\varphi_i$	Remaining stance time for mode sequences where lift-off occurs in $i^{\text{th}}$ clock cell
$\rho$	Forcing function (energy relating coefficient in the return map)
$\phi_i^*$	Mechanical phase in $\mathcal{E}_{i-1}$ coordinates at the event of switching from $(i-1)^{\text{th}}$ cell to $i^{\text{th}}$ cell
$M_i$	Affine phase transformation in $\mathcal{E}_i$ coordinates that occur during completely swept $i^{\text{th}}$ cell
$R_i$	Affine transformation to compute time to lift-off in $i^{\text{th}}$ cell
$\epsilon$	Loss term in energy component of the return map
$l_i$	Action term for the $i^{\text{th}}$ stiffness switch in the energy component of the return map
$T_{\mathcal{A}}$	Flight time in the $\mathcal{GAG}(e, N)$ mode sequence
$T_{\mathcal{G}}$	Stance time in the $\mathcal{GAG}(e, N)$ mode sequence
$\lambda_i$	$i^{\text{th}}$ eigenvalue of the Jacobian of the return map
$\text{tr}$	Trace of the Jacobian of the return map
$\Delta$	Discriminant of the Jacobian of the return map
$\overline{\text{tr}}$	A bound for trace of the Jacobian

$\overline{\Delta}$	A bound for discriminant of the Jacobian
$V(\alpha, s)$	Valid domain speed bound function

### Vector Valued Functions

$\tilde{\mathbf{f}}$	Physical hopper system function
$\tilde{\mathbf{f}}_{\mathcal{G}}$	Physical hopper stance mode system function
$\tilde{\mathbf{f}}_i$	Physical hopper system function in $i^{\text{th}}$ stance mode
$\tilde{\mathbf{f}}_{\mathcal{A}}$	Physical hopper aerial mode system function
$\tilde{\mathbf{c}}$	Ground collision map in physical coordinates
$\mathbf{c}$	Ground collision map in $\mathcal{E}_N$ coordinates
$\tilde{\mathbf{f}}^t$	Physical hopper state flow
$\tilde{\mathbf{f}}_{\mathcal{A}}^t$	Physical hopper state flow during aerial mode, $\mathcal{A}$
$\tilde{\mathbf{f}}_{\mathcal{G}}^t$	Physical hopper state flow during stance, $\mathcal{G}$
$\mathbf{g}_i$	Transformation from the physical coordinates, $\mathcal{X}$ , to the $i^{\text{th}}$ energy-phase coordinates, $\mathcal{E}_i$ .
$\mathbf{f}_i$	Hopper stance mode system function in the associated energy-phase coordinates, $\mathcal{E}_i$
$\mathbf{f}_i^t$	Hopper stance mode state flow in the associated energy-phase coordinates, $\mathcal{E}_i$
$\mathbf{h}_i^j$	Change of coordinates from $\mathcal{E}_i$ to $\mathcal{E}_j$
$\mathbf{r}$	Discrete-time Poincare map

$\mathbf{n}$	Coordinate transformation from $\mathcal{E}_N$ to $\mathcal{P}$
$\mathbf{G}_i^t$	Mechanical system flow in $\mathcal{E}_N$ during the $i^{\text{th}}$ ground mode, $\mathcal{G}_i$
$\mathbf{A}_i^t$	Mechanical system flow in $\mathcal{E}_N$ during the $i^{\text{th}}$ aerial mode, $\mathcal{A}_i$

### Matrix Valued Functions and Operators

$\mathbf{J}$	Jacobian of the return map
--------------	----------------------------

### String Valued Parameters

$\mathcal{G}\mathcal{A}\mathcal{G}(e, N)$	Abbreviation of the mode sequence $\mathcal{G}_{[1,e]}\mathcal{A}_{[e,N]}\mathcal{G}_N$
---	---

### Scalar Valued Parameters

$m$	The body mass
$g$	The gravitational acceleration
$\mu$	Normalized viscous damping coefficient
$\xi_i$	Normalized leg stiffness during $i^{\text{th}}$ cell
$\gamma_i$	Normalized natural oscillation frequency of the hopper mechanism during $i^{\text{th}}$ stance mode, $\mathcal{G}_i$
$\zeta$	Coefficient of restitution for toe ground collision
$N$	Number of cells in the output function
$T_c$	Clock period
$e$	Index of the cell where the lift-off occurs

$\phi_{TD}^i$	Mechanical phase in $\mathcal{E}_i$ at the touchdown
$\phi_{LO}^i$	Mechanical phase in $\mathcal{E}_i$ at the lift-off
$\beta_i$	Duration of the $i^{\text{th}}$ clock cell
$\beta_{[1,e-1]}$	Total duration of cells that are completely swept during stance

### Vector Valued Parameters

$\mathbf{o}$	Controller output shape vector
--------------	--------------------------------

## Feedback Analysis

### Scalar Valued Functions

$\tau(\nu, \alpha)$	Period tuning law
$\varphi_e(\alpha)$	The modified remaining time function
$\tau_{sync}(\alpha)$	Synchronization function
$\tau_{comp}(\nu)$	Composer function

### Scalar Valued Parameters

$T_n$	Nominal clock period describing the target behavior
$K_s$	Linear synchronization tuning law gain parameter
$K_c$	Sequential composition tuning law

# Open-Loop Controller Design

## Sets and Spaces

$\hat{\mathcal{D}}$  Approximate Poincaré delay state space

## Scalar Valued Functions

$\hat{b}_i^j$  Phase component of the approximate relating map,  $\hat{\mathbf{h}}_i^j$

$\hat{\phi}_i^*$  Approximate of the  $i^{\text{th}}$  switching phase,  $\phi_i^*$

$\hat{\varphi}_i$  Approximate of the remaining stance time function,  $\varphi_i$

$\hat{\epsilon}$  Approximate of the loss term,  $loss$

$\hat{l}_i$  Approximate of the  $i^{\text{th}}$  action term,  $l_i$

$\hat{\rho}$  Approximate of the forcing function,  $\rho$

err Scalar valued error term

$HOT_i$  Higher order terms for  $i^{\text{th}}$  approximate action expression

$L$  Cumulative action term

$\hat{L}$  Approximate cumulative action term

$\hat{\hat{L}}$  Basic cumulative action term

$\hat{\hat{\rho}}$  Basic forcing function

$\hat{\hat{l}}_i$  Basic  $i^{\text{th}}$  action term

$u$  Affine map relating the phase state,  $\angle$ , of the action space,  $\mathcal{L}$ , to the delay state,  $\alpha$ , of the Poincaré space,  $\mathcal{P}$

### Vector Valued Functions

$\hat{\mathbf{h}}_i^j$  Approximate relating transformation from  $i^{\text{th}}$  ground mode,  $\mathcal{G}_i$ , to  $j^{\text{th}}$  ground mode,  $\mathcal{G}_j$

$\mathbf{l}_i$  Vector representation of  $i^{\text{th}}$  approximate action function,  $\hat{l}_i$

### Scalar Valued Parameters

$x_1^{\text{apex}}$  User-specified target physical hopping apex height

$A_i^j$  Amplitude coefficient for the energy relating map  $a_i^j$

$A_i$  Action function amplitude coefficient for  $l_i$

$\delta b_i^j$  Maximum error in the phase component of the approximate relating map,  $\hat{b}_i^j$

$S_i$  Phase shift for the approximate  $i^{\text{th}}$  action,  $\hat{l}_i$

$\delta \phi_i^*$  Max approximation error in  $i^{\text{th}}$  switching phase approximate,  $\hat{\phi}_i^*$

$L$  Number of levels in triangle portion of triangular schedule,  $\psi(\theta)$

$\bar{\beta}$  Duration of the spike portion of the spike shape function

$c$  Ratio between natural oscillation frequencies of two consecutive levels of the spike shape function

# Generic Spring Potentials

## Scalar Valued Variables

$\phi$	Hip angle state for VWSM (the angle between the vertical and leg in counter clock-wise direction)
$h$	Total mechanical energy value

## Scalar Valued Functions

$\tau$	Hip actuator torque output
$\sigma$	Kinematic relationship relating the vertical height, $x_1$ , to the hip angle, $\phi$
$F_\xi^s$	Generalized one parameter family of spring forces
$F_\mu^d$	Generalized one parameter family of damping forces
$\Delta H$	Generalized relative change in total energy
$\Delta\phi$	Generalized relative change in mechanical phase

## Vector Valued Functions

$\mathbf{f}_\xi$	Generalized one parameter family 1-DOF mechanical system dynamics in the physical coordinates, $\mathcal{X}$
$\mathbf{f}_\xi^t$	Generalized one parameter family 1-DOF mechanical system flow in the physical coordinates, $\mathcal{X}$

## Scalar Valued Parameters

$l$	Length of the rigid leg
-----	-------------------------

$s$  Measure of softness for a spring potential law

# CHAPTER 1

## Introduction

### 1.1 Motivation and Scope

Terrestrial locomotion has always been important for mankind. Dexterous and efficient travel over unstructured terrain is invaluable for not only military but also many commercial applications. Legged locomotion offers unparalleled mobility in a wide array of natural settings. However, the design, implementation and control of legged machines have each proved to be very hard engineering problems. Almost no robotic platforms have been able to operate outside carefully controlled environments let alone outperform their biological counterparts.

Recent collaborations between biologists and engineers [2, 3] have led to a novel class of hexapod robots [4, 5] that are among the very few robotic systems that can successfully negotiate unstructured environments. In particular, RHex [6], a computationally and energetically autonomous robotic platform designed and implemented as a part of the CNM project [2], can locomote over various terrains with surprising agility. Over flat ground its dynamical tripod running gait achieves specific resistances as low as 0.6 [7] and its top speed exceeds five body lengths per second (2.5 m/sec) [8]. More importantly, it can handle highly broken and unstable surfaces

at one body length per second [9]. In addition to the basic tripod gait, the versatility of legs allows RHex to perform a number of other tasks, such as climbing stairs [10], bounding [11], pronking [12], flipping [13], leaping [14], and even running bipedally [15].

The majority of legged robotic platforms locomote in quasi-static and slow operating regimes [16, 17]. Their locomotion control is usually performed by strongly feedback driven high gain actuation [18–20] that does not lend itself to efficient behavior. It is implicitly assumed that all degrees of freedom must be under control which leads to complex mechanisms [21].

In contrast, RHex, and robots of its class, adopt a drastically different approach to control of locomotion. RHex [6] employs minimal active actuation — one motor per leg — and relies on passive dynamics of its carefully designed morphology to stabilize its locomotion. Another important distinction is that the majority of RHex’s behavioral controllers either operate in open-loop or use very low bandwidth sensory feedback. Hence, the center of mass motion of RHex is not explicitly dictated but it emerges from the interactions between the open-loop excitations and the environment through the compliant legs [22]. The mechanical and algorithmic simplicity of open-loop controllers have been the primary enablers underlying the robust and durable design of RHex. We will discuss the role of complexity in the context of physical implementations and defend the foregoing opinion in Section 1.3.1.

The RHex platform allowed us to accumulate abundant empirical data and develop a certain level of intuition on how to design and implement open-loop controlled dynamical behaviors. However, we still lack a systematic methodology to guide the design process and verify performance of the limit behavior. In the robotics community there has been a sizable interest in open-loop control strategies that we will review in Section 1.3.3. Unfortunately, most of this earlier work primarily focuses on

demonstrating the feasibility of open-loop control on a case by case basis. To date a significant portion of the behavioral development for RHex (as well as most open-loop controlled physical devices) remains more or less a black art where the developer's intuition is the sole design tool.

Unarguably, formal design tools, which offer solutions with predictable properties, can lead to groundbreaking developments in a field. A good example is how Bode and Nyquist transformed the analog filter design process [23]. Furthermore, a systematic framework can allow a much larger community to effectively address hard problems. More importantly, formal insight can lead to discoveries that are not possible or very improbable to achieve by simple intuition.

This thesis presents a set of analytical results toward establishment of a formal framework to design and verify open-loop controllers for rhythmic tasks. Our work is primarily motivated by control of dynamical legged locomotion. However, the reader should note that the majority of these ideas seem also applicable to a wider variety of other rhythmic tasks as well.

In Section 1.2 we will first present a review of the relevant biological discoveries on animal locomotion control which served as guidelines in our work. This will be followed in Section 1.3 by a review of the locomotion control literature where we will divide the discussion into three parts according to the level of sensory feedback usage. Finally, we will summarize the contributions of this thesis in Section 1.4. Our discussions of the open-loop controllers will be organized around two aspects of the open-loop controller design process: 1) design of a passive mechanical system with favorable natural dynamics; and 2) design of an effective excitation scheme.

## 1.2 Biological Inspiration

### 1.2.1 Terrestrial Locomotion

In simplest terms, terrestrial locomotion is the displacement of a body from one place to another over a land mass. In terrestrial locomotion, the motion of the center of mass is governed by two forces: 1) the gravitational pull; and 2) the ground reaction force. Obviously, the former is not a controllable entity leaving the latter as the sole source of control. Hence, in the most generic sense, terrestrial locomotion control can be defined as the regulation of the ground reaction force using the available actuators within the locomotor.

The search for devices that can make locomotion easier has been an ongoing effort for mankind. Since its invention circa 4000 B.C. the *wheel* has been mankind's primary terrestrial transportation technology [24]. One immediate advantage of the wheel is its mechanical simplicity. However, the primary driving force behind the widespread use of wheels is the straightforward control authority it offers over the ground reaction force, which in turn allows simple and computationally cheap control algorithms to effectively control wheeled vehicles.

Unfortunately, efficient wheeled locomotion requires a certain environmental condition, namely the availability of a flat surface [25], which is seldom satisfied by natural terrain [26]. For efficient transportation wheeled vehicles require another less mentioned human invention, the road infrastructure, whose construction is an expensive and inflexible commitment. As a result, the operational domain of wheeled vehicles is strictly limited to a small percentage of the world's land mass [20].

Despite its ubiquitous presence in man-made artifacts, wheels seldom appear in biological systems, and only at very microscopic scale [27]. There are certain instances of animal locomotion that resemble wheel-like locomotion for an outside observer [28].

However, closer inspection of these operating regimes reveals that the ground reaction force profiles do not match that of wheeled locomotion.

Instead, Nature employs a radically different locomotion technology based on legs. There are a number of evolutionary and developmental reasons why animals prefer legs instead of wheels [29]. Legs offer a more suitable solution for the real world terrestrial locomotion problem [30]. Unarguably, legged animals display unmatched agility over highly broken and unstable terrain that no man made vehicle can even approach.

Unfortunately, the performance of legs does not come for free. The control of legged locomotion is many orders of magnitude harder than that of wheeled vehicles. This can be attributed to the unique properties of legs. First, a leg has a limited range of motion, and therefore, to maintain a continuous body motion supporting legs must go through alternating stance and aerial phases. As a result, the hybrid nature of a single leg can only offer intermittent control authority over the ground reaction force. Second, a single leg is a typically under-actuated mechanism. Particularly for mechanisms, where the center of mass is high above the ground, forces and torques generated at the ankle — the joint where the leg makes contact with the ground — have little affordance over the motion. To overcome this problem a number of humanoid robots [31, 32] use large feet, but ankle torques remain severely limited. Hence, the ankle is often modeled as a free rotating joint [33, 34].

It appears that neither leg nor wheel immediately offers a perfect engineering solution for the terrestrial locomotion in natural settings. The ideal locomotion technology would be the one which can be controlled as easily as wheeled vehicles while providing the performance of legs. The holy grail of the terrestrial locomotion is a design methodology that can recover the ease of control and implementation robustness of legged machines without losing performance.

## 1.2.2 Functional Biomimesis

Over hundreds of millions of years evolution has led to the development of specialized neural, muscular and skeletal systems for animal locomotion [35] that are incredibly effective in handling real world problems using realistic resources. Therefore Nature, offering a large number of working examples, serves as an invaluable resource in the quest to develop effective control strategies for legged locomotion.

In fact, documented interest in animal (and in particular human) locomotion dates back to ancient Greece [36,37] and Rome [38]. The first mechanical legged machines, such as Leonardo da Vinci’s artificial lion [39] and Jacques de Vaucanson’s mechanical duck [40], were mere imitations of Nature. The first analytical investigation of legged locomotion in humans [41] appeared in the Renaissance, leading to systematic studies on human gaits [42–44] in the 19th century that established the basis of modern biomechanics. Numerous developments in many fields of basic science and technology in the 20th century, particularly the widespread use of computers, paved the way to further experimental studies of ever increasing detail and complexity. By the end of the 20th century, scientists had collected a large amount of data on the neural and mechanical aspects of animal locomotion.

It is often argued that robotic implementations that mimic natural systems, as in the biology-as-default approach [30], can demonstrate a comparable locomotion performance. Unfortunately, blindly copying nature often leads to impractical designs [45]. The failure of mimicry can be attributed to two basic reasons. First, natural evolution is a shortsighted process that works on the “just good enough” principle [29]. The biological systems we see all around us are not optimal but just satisfactory solutions. Moreover, these designs carry details to satisfy other biological needs, constraints to which the engineer is not necessarily bound. Second, the building blocks of artificial and biological systems have very different characteristics [46].

While animals are made out of compliant and soft tissues that use complex chemical processes to generate propulsion, engineering solutions choose rigid contraptions and electrical motors. These differences make mimicry, if not impossible, a very hard task at best.

Despite the significant differences between nature and engineering, studies of biological systems can still provide useful information for the engineering design process. The key to success is the identification of fundamental design principles that can be realized using available engineering tools. “Functional biomimesis” [47] driven by the discoveries of integrative biology [48] outlines such a design methodology. Recent years have witnessed the development of successful novel robotic platforms [4, 5] that demonstrate the effectiveness of functional biomimesis. Our work in this thesis is in large part inspired by the success of the control principles employed in these robotic platforms.

The most important outcome of such biological studies is the identification of fundamental structural and algorithmic aspects of locomotion control in animals. These discoveries offer novel solutions to engineers for the difficult legged locomotion control problem. We classify these biological observations into two categories: 1) observations concerning the controller structure; and 2) observations concerning mechanical properties.

A great number of biological studies have focused on motor nervous systems — the biological locomotion controllers [49]. In these studies, biologists have demonstrated that the origins of the rhythmic motor signals reside in central pattern generators (CPGs) [50] — oscillatory neural primitives that are ubiquitously found in both vertebrates [51] and invertebrates [52]. In essence, a CPG is a nonlinear tunable oscillator [50] whose periodic output is fed to a related muscle group as the motor excitation signal. Although a CPG is capable of generating stable motor patterns in the

total absence of any sensor feedback or command from brain [53], in an intact animal, the output of a CPG is strongly influenced by the sensory data and the descending commands from the brain as well as the neighboring CPGs [54–57]. CPGs have been studied in experimental [53] and in mathematical inquiries [58–62]. To understand the working principles researchers have developed models of biological CPGs with varying levels of detail.

To accommodate a wide variety of tasks animals have evolved very high degree of freedom morphologies (for instance, a cockroach has 72-DOF [2]) that are highly redundant for the task of locomotion [63]. Although animals are capable of moving their limbs in extremely diverse ways, they typically choose to move in a very stereotypical manner [64,65]. Coordinated movement of the limbs effectively collapses the dimensionality of the body dynamics. In fact, biomechanical research on animals has demonstrated that independent of body morphology and size [66] the center of mass motion is successfully described by low-dimensional inverted pendulum (walking) and spring-mass (running) models [67]. Biologists attribute the coordination of limb movement to a network of coupled CPGs that generates synchronized motor commands [68]. Coupled oscillators present an elegant and scalable framework to describe and generate synchronized actions. Hence, in recent years the coupled oscillators concept has received a great deal of attention in engineering [69], mathematics [59,70–72] and biology [62,70,73].

The motor nervous system is a heterarchical network of CPGs and sensors [54]. In [47] we introduced a useful classification of controllers based on two properties: 1) sensor dependency; and 2) the degree of centralization in the control implementation. The former is defined by the coupling strength between CPGs and sensors. The latter is characterized by the coupling structure and strength among physically distributed CPGs. Locomotion control strategies vary considerably across different species as

well as across different behaviors within the same animal. It is possible to find examples that lie at seemingly any location within the two dimensional controller space defined by the feedback/feed-forward and distributed/centralized attribute axes. For instance, the death-head cockroaches employ a centralized feed-forward control strategy during fast running [74], whereas stick insects, a very slow moving six legged animal, employ a distributed and strongly feedback driven control mechanism [75].

Animals have to operate within the confines of the numerous fundamental trade-offs imposed by the physical world. In order to achieve high energetic efficiency and control performance animals aggressively exploit their passive dynamics [76–78]. The passive compliance of limbs appears to be the basis of energetically efficient locomotion [79]. Moreover, the configuration of the limbs and body can lead to passive stabilization of locomotion [48, 80]. Therefore, it can be concluded that the passive properties of the mechanical system play a significant part in locomotion control [81].

Another important aspect of mechanical design is the actuator dynamics. In engineering practice, actuators typically take the form of torque and force sources. However, in biological systems motor recruitment signals appear to modulate some mechanical parameter of the underlying musculoskeletal structure, such as the effective compliances of the joints. Namely, under static conditions, a fixed level of muscle activity defines a unique speed-force-extension profile for a muscle [82]. This is also supported by the cumulative activation of muscle units as a function of the motor signal strength [83]. Further evidence from biomechanical research on human locomotion shows that effective leg spring stiffness is controlled by the motor nervous system in order to adapt environmental conditions [84–86].

## 1.3 Control of Legged Locomotion

There is a rich literature on the design and control of legged robots. The success of a physical implementation appears to be strongly correlated with the complexity of its design which, in turn, depends on the level of sensory feedback utilization. In Section 1.3.1 we will discuss this relationship in more detail. In the robotics literature, the dependence upon sensory feedback seems to fall into three natural groupings which we will individually review in Section 1.3.2, Section 1.3.3 and Section 1.3.4.

### 1.3.1 Complexity must be Reduced

Any physical system is a collection of tightly coupled components that coexist and cooperate to achieve a specific goal. The number of components and their inter-relationships collectively determine the complexity of the design. The components of a typical control system can be classified into four groups: sensors; actuators; processors; and passive mechanical parts. Introduction of each component unavoidably increases the design complexity.

Sensors collect physical measurements from the mechanism and its environment. Often these raw measurements are not immediately useful and need to be processed to extract information pertinent to the task at hand. Unfortunately, there are physical limitations in both the sensor as well as the processing that introduce noise into the readings.

Actuators are active components embedded within the passive mechanical structure that perform physical work. All actuators are limited in power. This is a fundamental constraint that plagues the performance of the behavior if the control algorithm does not take this aspect of the actuators into consideration.

Processors are the programmable units in the control system. They gather infor-

mation from the sensors and compute control actions according to user commands or task specifications. Although available computational power is increasing exponentially there are still limits, and therefore, the implementation of control algorithms must be pursued accordingly. The parts of a control system are organized within a structure. The hierarchical and distributed nature of the physical controller implementation imposes certain interfacing issues on the physical parts.

Naturally, all parts need to exchange information in one form or another. This exchange requires a medium - a communication network. No matter what this medium is, physical communication channels impose a bandwidth limitation, which restricts the amount of information exchange between parts.

For a physical system an assessment of its practical value can be based on its durability and robustness. Durability represents the capability to withstand wear and tear or decay, which are unavoidable in the physical world. Robustness is a measure of how well a particular system operates in the face of such unexpected conditions. Unavoidably, complexity of design incurs detrimental effects for both aspects of the system performance. A simple design allows for easy development where the problems caused by human factors as well as components failure are minimized. Hence, a sound engineering design process aims to find the simplest solution for a given problem.

### **1.3.2 Feedback Control**

The origin of the unique challenges arising in legged locomotion control is the limited and intermittent control authority that a single leg can offer. Especially in dynamical operating regimes, where the body goes through periodic aerial phases, the actions of the locomotion controller must be well synchronized with the motion of the mechanical leg. At first glance this dependency suggests that stable legged locomotion can only be achieved by a strongly feedback driven control strategy. Indeed, the

majority of the control strategies for intermittent rhythmic tasks are strongly feedback driven. This section will review a relevant subset of the feedback control literature and point out advantages and shortcoming of this approach.

The complexity of control underlying legged locomotion can be resolved by restricting operation such that the body is supported by multiple properly configured legs at all times. In fact, most legged machines [16, 17, 20] have adopted this strategy to eliminate the need for active stabilization of the body. Furthermore, the closed kinematic chain formed by the supporting legs offers complete control authority over the center of mass motion, allowing designers to use classic robotic control methodologies [20], such as inverse dynamics control [87]. However, the consequent limitation to quasi-static operation severely diminishes the efficacy of this paradigm. Moreover, the associated stiff and high-gain actuation completely precludes any passive energy recovery that would improve energetic efficiency of locomotion.

In his groundbreaking work [88], Raibert demonstrated that dynamical locomotion can be achieved by exciting passive dynamics in synchrony with a robot's motion. His approach divides the control problem into three parts: 1) vertical hopping; 2) fore/aft speed; and 3) pitch. Although these degrees of freedom are coupled by the dynamics of the mechanism Raibert's control methodology handled them in a decoupled manner.

Raibert's control of vertical hopping takes the form of an energy regulation mechanism in which the controller detects the onset of the decompression phase and applies a constant force for a constant duration to replenish the energy dissipated over the past cycle due to mechanical losses. This control strategy was studied extensively in the robotics literature [89–92] and shown to have globally asymptotically stable limit behavior under certain conditions. Linde demonstrated a successful application of a very similar strategy to the control of a pendular motion [93]. There have been a number of variants of Raibert's original energy regulation technique. In the bow

legged monopod [94] Zeling and Brown chose to trigger thrust with the detection of touchdown. In ARL monopod II, the most efficient legged locomotor of its time [95], Ahmadi and Buehler implemented an alternative energy regulation technique where the rest length of the leg spring is continuously varied as a function of the body states [96].

Raibert observed that the touchdown angle and the limit fore/aft speed are monotonically related in SLIP. Based on this observation Raibert implemented a simple controller that corrects the touchdown angle according to the error in the fore/aft speed. As in the vertical hopping control this intuition-driven control strategy was later analytically investigated [97]. Schwind et al. proposed an alternative feedback controller leveraging their analytic insight on the 2-DOF SLIP dynamics [98].

The strong descriptive power of relatively simple dynamical models such as SLIP led to the idea that these models might also be used as control targets. The ability to encode a desired task by a low dimensional dynamical model with minimal complexity has paramount value and is the basis of “template” based control [99]. The first instantiation of this ideology can be found in the multi-legged robots of Raibert [100]. Another example of target dynamics based control is the virtual model scheme [101, 102] where the forces and torques given by the virtual mechanical components are projected back to the underlying mechanism to compute the control commands. Saranli et al. presented a series of analytic and numerical studies in which simple dynamical models are “anchored” into high degree of freedom mechanisms [33, 103]. In their recent work [?, 34] Grizzle and Westervelt described a controller for biped walking through properly tuned zero dynamics, also based on a similar principle. Of course, template based control is not limited to legged locomotion control. Nakanishi et al used a simple pendulum model to control a brachiating robot [104].

Juggling — a dynamical intermittent behavior that is analogous to legged loco-

motion — was also investigated by several researchers. A robust feedback control algorithm, a “mirror” law, for planar juggling was proposed by Buehler in which the paddle motion is kept hostage to the position of the ball. He successfully demonstrated one [105] and two ball [106] juggling in a physical setting. Later, his work was extended to the spatial setting [107–109].

Theoretically, feedback control typically offers good disturbance rejection and fast convergence to the limit behavior since the control actions are strongly coupled with the state of the task being controlled. However, physical implementation of feedback controllers presumes a considerable sensory infrastructure. In some cases the sensor can be an basic event detector or a simple estimator [110]. However, some physical settings require a more elaborate mechanism to access the desired information [111]. As the task is more intimately related to the environment sensors may even constitute the majority of the controller complexity [112, 113]. Often the required task information, such as the position of the juggling ball, is not immediately available and must be estimated. This estimation process requires a system model, which may not be easily accessible. Naturally, estimation also introduces a considerable computational load. Moreover, implementation of sensor hardware is a very cumbersome process. Physical limitations of hardware and the approximate model used by the estimation process introduce sensory noise, which degrades the control performance.

Hence, physical feedback controllers tend to be very complex mechanisms. This immediately reflects on the robustness and durability of the overall system. In some cases, the introduced design complexity can be prohibitively high.

### 1.3.3 Open-Loop Control

#### What is open-loop control?

Sensory information may be classified into two groups: proprioceptive; and task-level. The former information concerns the mechanism itself and is not sufficient to deduce the state of the task. The latter captures the state of the behavior under control. For instance, an encoder of at the hip joint is a proprioceptive sensor, which measures the configuration of the associated leg. However, to obtain the speed of locomotion a more elaborate task-level sensor is required [114–116].

In the simplest terms open-loop control is the regulation of a task in the total absence of task-level information. An open-loop controller is in essence a signal generator. Its output, which we will refer as the “excitation,” drives the actuators, embedded within the underlying passive mechanical system. The target behavior emerges from the interactions between the resulting time-varying system and its environment [55,88]. In these settings, passive dynamics play a crucial role. Therefore, the design of open-loop control must include: 1) an effective mechanical actuation mechanism; and 2) a proper excitation signal.

Our discussions will focus on legged locomotion control. However, the reader should note that these ideas are also applicable to a much wider spectrum of cyclic tasks. Inspired by the structure of the biological motor nervous system, discussed in Section 1.2.2, we posit a generic open-loop controller consisting of an oscillator and a scalar “shape function.” The oscillator, also referred to as the “clock,” generates a periodic phase signal. Note that the clock is the abstraction of the CPG in biological systems. The output of the open-loop controller, that we call the excitation signal, to be fed to an actuator is a scalar map of the periodic clock phase through the shape function, which defines the detailed shape of the output.

## **What are the advantages and disadvantages of open-loop control?**

Since open-loop controllers do not employ any sensory data, their physical implementations do not include any task-level sensor modalities and the related hardware and software infrastructure. Furthermore, open-loop control algorithms are typically computationally cheap and easy to implement. The resulting design simplicity immediately improves the durability of the overall system and decreases the cost of design and manufacturing. Moreover, since the control algorithm does not employ any sensors there is no sensor noise issue associated with the controller.

Unfortunately, there is no established design methodology for open-loop controllers. The state-of-the-art in open-loop controller design heavily relies on the researcher's intuition to determine the actuation mechanism and the generic form (but not the exact shape) of the excitation signal. In order to achieve successful behavioral control a great deal of effort is typically spent on optimization of the excitation signal [8].

Those aspects of open-loop control, which make it attractive, are also the sources of its shortcomings. A practical system must be able to handle a wide range of external disturbances and rapidly converge to the limit behavior. However, the lack of task level sensing often results in a very small basin of attraction. In this case, the limit behavior has a very small operational domain and is very sensitive to external disturbances. Moreover, the speed of convergence to the desired behavior is usually slow.

## **Related Work**

A number of researchers have looked into simple abstract settings to get better analytical insight into those conditions that render open-loop control a viable option for regulation of rhythmic tasks. Although these models differ in detail, they share a

common structure, which is composed of a 1-DOF Newtonian mechanical system and an open-loop controller generating a periodic excitation signal that modulates some physical parameter of the mechanical system.

**Clock Driven 1-DOF Hopper** The hopper presents the simplest mechanical model that captures the fundamental features of a typical dynamical gait of a legged platform. The classical 1-DOF hopper model consists of a point body mass riding on a dissipative compliant leg. The body is constrained to move vertically under the influence of the gravity. The leg intermittently supports and propels the body upward. The control input takes the form of an adjustable spring parameter. The controller is a periodic signal source. Ringrose [117] and Berkemeier [118] have investigated the clock driven 1-DOF hopper model. They chose the rest length as the control input to the hopper and their excitation signal is limited to a pulse train that alternates between a normal and a forced rest length. Their work primarily focus on the demonstration of the existence of open-loop controlled stable operating regimes. Both researchers present detailed numerical and empirical studies but only limited analytic insight. Moreover, they offer no formal design guidelines. Ringrose considers an approximate and partial return map for his analytic work. Berkemeier computes a full return map using perturbation methods. Both analyses assume small damping and small changes in rest length. Chapter 2 will present a detailed analytic study of a slightly different 1-DOF hopper that is driven by a more generic clock controller. Our major contributions from this analysis will be summarized in Section 1.4.1. The excitation schedule employed by both Berkemeier [118] and Ringrose [117] is an instance of our generic setting which is considered in Section B.4.1.

**Clock Driven 1-DOF Juggling** Juggling presents a simple dynamical intermittent task and has received considerable attention from robotics researchers [119–121]

Furthermore, juggling is also relevant to the study of legged locomotion control since it is an analogous problem to hopping in which the actuator is located in the environment instead of on the robot. The classical 1-DOF juggling model consists of a point mass ball (the robot) and a rigid, flat, infinitely massy paddle (the environment). The body is constrained to move along the vertical. Its collisions with the paddle are modeled to be plastic with a coefficient of restitution that defines the change in the energy of the ball upon each collision. The paddle, whose vertical motion is dictated by the output of the clock controller, is not affected by these collisions. The return map of a clock driven 1-DOF juggler has a simpler structure than that of the 1-DOF hopper since a memoryless (algebraic) map governs the contact dynamics. Holmes identified the period one limit behavior and demonstrated the bifurcations of the coupled behavior [120]. Schaal studied a number of juggling tasks in [121] including the classical 1-DOF juggling. He chose to move the paddle according to a sinusoidal trajectory, which led to a family of locally stable period-one behaviors. The basin of attraction was computed numerically. He also demonstrated in empirical studies that the open-loop control is a feasible alternative to feedback controllers in [105,122]. Swanson [119] showed that the shape of the excitation signal, the trajectory of the paddle, has a paramount effect on the stability properties of the coupled behavior. His carefully designed parabolic trajectory gave rise to a globally asymptotically stable period-one limit behavior.

### **1.3.4 Adaptation of Open-Loop Control**

#### **Adaptation of the Open-loop Excitation**

A large number of robotic control algorithms assume the availability of high bandwidth sensory information. For reasons explained in Section 1.3.1 such strongly feedback driven controllers are plagued by the trade-offs of the physical world. One axis

of the controller design space captures the level of centralization of the control [47]. The other axis determines how strongly the output of the controller depends upon sensory feedback. For every physical task there is a sweet spot along the sensor dependency axis where the advantages and disadvantages of sensor dependency are balanced [47, 123].

Despite their various shortcomings open-loop controllers [121] offer a very favorable starting point for effective low bandwidth closed-loop controllers. The stability properties of the limit behavior can be considerably improved by periodically (or continuously) correcting the shape and timing of the feed-forward excitation signal according to a rule based on the states of the underlying mechanism [117, 118]. The effective feedback bandwidth of discrete-time updates to an underlying open-loop signal generator is typically much lower than continuous-time feedback. When the bandwidth of the sensory feedback is low enough one is tempted to use the term “adaptation” rather than feedback [117, 118]. In [110] we presented an adaptation scheme for the hexapod robot RHex that modifies the excitation scheme, the leg trajectory, according to the estimated surface slope such that walking over inclined surfaces improve. In a more recent work [7] a more elaborate profile adaptation law improves rough terrain handling.

Direct biological evidence for such “adaptation” strategies is provided by [124] which shows that tonic electrical stimulation of sensory nerves in the thoracic ganglia of locusts increases the wing beat frequency. Another supporting observation comes from dogfish studies. The frequency of swimming decreases as the extent of sensory feedback is decreased by chemical injections [57].

**Adaptation in 1-DOF Hopper Control** Both Ringrose [117] and Berkemeier [118] identified the shortcomings of open-loop control and concluded that in physical

applications the open-loop excitation signal needs to be adapted according to the state of the mechanism. They chose to vary the period of the excitation signal once in every hop according to the timing of a mechanical event such as touchdown. The particular period alternations synchronize the clock with the motion of the hopper such that the motion of the hopper is better influenced by the clock driven modulations of the rest length. Buehler et al. demonstrated that stable dynamical walking in a quadruped can be controlled by simply triggering the timing of the excitation signal [125]. In recent work [126] Cham implemented a period adaptation controller for hexapedal robot Sprawlita which lead to significant improvement of the disturbance rejection. However, none of these previous studies offer much, if any, analytical insight as to why these particular adaptation mechanisms result in improvement in behavioral stability. In Chapter 3 we present a discussion of the discrete-time adaptation of the clock period. A summary of our contributions can be found in Section 1.4.2.

**Continuous Coupling Between the Mechanical System and the Clock** Inspired by the heterarchical structure of the motor nervous system [54] where the CPGs and sensors are coupled via bi-directional pathways, Hatsopoulos [55] investigated continuous-time coupling between the clock controller and a 1-DOF cyclic mechanism. The details of Hatsopoulos’s model are considerably different and much more complex than that of the classical hopper. His model consists of a 1-DOF linearized pendulum actuated by a torsional spring and a Van der Pol oscillator acting as the clock controller. The spring constant of the torsional spring is adjustable and driven by the output of the clock. The nominal period of the oscillator is assumed adjustable and continuously modulated by the configuration of the pendulum. The complexity of the model dynamics prevents Hatsopoulos from achieving a substantial formal analysis of the coupled behavior. He numerically demonstrates that this bi-

directional coupling leads to synchronization of the mechanical system and the clock phases. There can be found successful physical implementations of continuous coupling. Weingarten et al. describes a feedback driven locomotion controller for RHex in [7] where the parameters of the standard open-loop tripod gait controller (a clock controller) are adjusted according to the sensory data.

## **1.4 Contributions of This Thesis**

### **1.4.1 A Novel Clock Driven 1-DOF Hopper Model**

We present a detailed analysis of a novel clock driven 1-DOF hopper model, differing from those previous in two major regards. First, we use spring stiffness as the control input to the mechanical hopper. This setup offers a considerable analytical simplification leading to our detailed mathematical results. Second, our clock controller takes a considerably more generic form addressing any arbitrary piece-wise constant periodic excitation signals.

The primary contribution of this thesis is the identification of a set of sufficient conditions that lead to local asymptotic stability of the forward coupled system. Our results indicate that a dissipative hopper driven by a properly designed periodic excitation signal demonstrates locally stable hopping behavior. Furthermore, we analytically demonstrate that the desired task, the apex height, is encoded by the clock period independent of the properties of the mechanical hopper.

### **1.4.2 A Family of Adaptation Mechanisms**

The analytical insight into the forward coupled system gives rise to a family of stride-to-stride clock adaptation mechanisms. We discuss two classes of adaptation laws: 1) “synchronization laws” which improve the local convergence speed; and 2)

composition laws which enlarge the size of the basin of attraction. The synchronization adaptation laws only require the phase difference between the controller clock and the mechanical hopper. This task-level sensing can be implemented by simply detecting the touchdown event. The closed loop system exhibits considerably faster convergence to the behavior and has a larger basin. The composition laws enlarge the basin by back chaining [109]. In the open-loop analysis we discovered that the location of the fixed point and its basin are parameterized by the clock period. This allows us to displace the basin to capture the states of the coupled system and pull it gently towards the desired goal. The implementation of the top layer requires the measurement of the touchdown speed, which can be estimated based on the flight time. The resulting closed loop system has a much larger domain of attraction. Furthermore, the same back-chaining strategy allows us to encode and control more complex tasks in which the instantaneous target apex height needs to go through scheduled variations.

### 1.4.3 Extensions of the Basic Results

The formal results in Chapter 2 offer invaluable insights into the working principles of open-loop control of rhythmic tasks, which in turn inspire a number of extensions — each aiming to bring these preliminary abstract results one step closer to concepts and processes that can offer practical solutions to real-world control problems.

We present a detailed discussion of open-loop controller design in Appendix B. Leveraging the basic results from the analysis of the clock driven hopper we propose a computational design algorithm to identify open-loop controller parameters that render a user-specified hopping task locally asymptotically stable. The proposed approach is based on a set of approximations which allow us to relate the controller parameters to the existence of stable operating regimes. At its current state the design

process does not offer guarantees of stability and it yields limit behavior that is close but not identical to what the user specifies. Yet, our numerical studies suggest that for a wide spectrum of physically-relevant tasks the procedure successfully constructs open-loop controllers that yield limit behaviors that are close to the user-specified tasks.

Appendix C presents a second extension to the basic analysis in which we investigate the role of the mechanical system in an open-loop control setting. For this part of our work we consider a family of 1-DOF Lagrangian systems that are modeled as lossy nonlinear springs. Our numerical studies suggest that certain spring potentials are suitable for certain tasks. Furthermore, the importance of mechanical dissipation, a result from the study of the linear prismatic hopper, persists in this generic setting. The long term goal of this study is to offer guidelines for the design of the mechanical structure such that the resulting system offers advantageous dynamics for open-loop control.

## CHAPTER 2

### A Clock Driven 1-DOF Hopper

#### 2.1 The Model

##### 2.1.1 A Mechanical Hopper

The mechanical hopper, depicted in Figure 2.1, is a 1-DOF Lagrangian system consisting of a point mass body constrained to move vertically and a prismatic massless leg which intermittently supports and forces the body. Due to the analytical advantages it offers we study the dynamics of the hopper in a dimensionless setting whose derivation can be found in Appendix A. In this dimensionless coordinate system,  $\mathcal{X}$ , the leg is modeled as a lossy, zero rest length<sup>1</sup>,  $\bar{x}_1 = 0$ , Hooke's law spring with tunable spring constant,  $\xi \in \mathcal{K} := (\mu, \infty)$ . The sole control input to the mechanical hopper will be this adjustable spring constant,  $\xi$ . We model mechanical energy dissipation by two loss mechanisms: 1) plastic ground collisions at touchdown parameterized by a “collision restitution” coefficient,  $\zeta \in (0, 1]$ ; and 2) constant viscous leg damping of magnitude  $\mu \in \overline{\mathbb{R}^+}$ . For the sake of simplicity we will assume that the collision restitution coefficient,  $\zeta$ , and the damping coefficient,  $\mu$ , are related by

---

<sup>1</sup>Note that the actual rest length is not zero but the dimensionless coordinate transformation in Appendix A shifts the rest configuration to the origin.

$$\zeta = \exp[-\mu] \quad (2.1)$$

which allows us to characterize the overall mechanical dissipation characteristics of the hopper by the viscous damping coefficient,  $\mu$ .

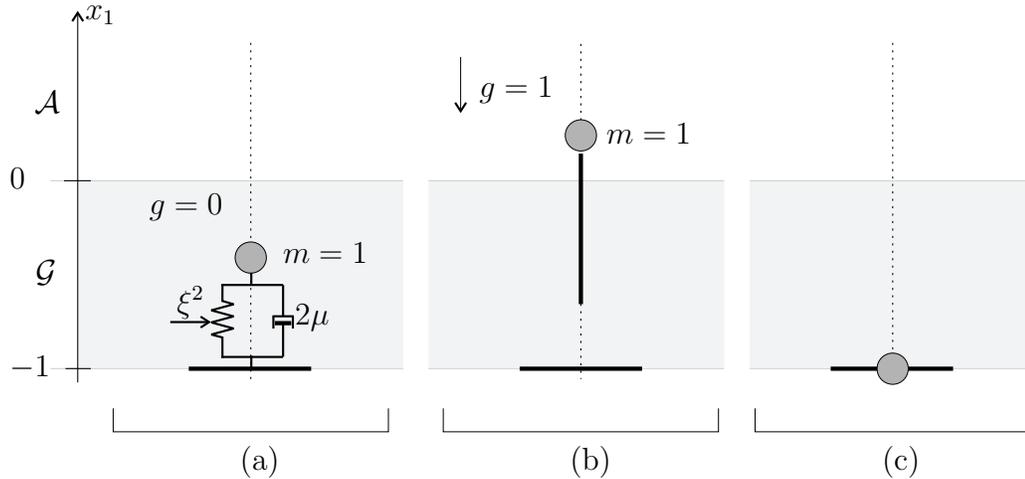


Figure 2.1: Mechanical hopper (a) in stance phase; (b) in aerial phase; (c) at ground collision.

We let the set of physically valid body heights,  $x_1 \in [-1, \infty)$ , be those configurations where the body is above the ground surface which is located at  $x_1 = -1$  in this dimensionless setting. Accordingly, we define the “physical coordinate system” for the mechanical hopper,  $\mathcal{X} := [-1, \infty) \times \mathbb{R} - \{0\}$ . The second coordinate,  $x_2 = \dot{x}_1$ , is the body velocity. We exclude the origin where the trivial solution resides since this analysis only concerns cyclic limit behaviors.

The nonlinear *hybrid* dynamics of the hopper,

$$\dot{\mathbf{x}} = \tilde{\mathbf{f}}(\mathbf{x}, \xi), \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathcal{X}, \quad (2.2)$$

has two modes: a “stance mode,”  $\mathcal{G}$ ; and an “aerial mode,”  $\mathcal{A}$ . Correspondingly,

the state space of the hopper,  $\mathcal{X}$ , is partitioned into five cells: a stance set,  $\mathcal{X}_{\mathcal{G}} := \{\mathbf{x} \in \mathcal{X} | x_1 < 0\}$ ; an aerial set,  $\mathcal{X}_{\mathcal{A}} := \{\mathbf{x} \in \mathcal{X} | x_1 > 0\}$ ; a lift-off set,  $\mathcal{X}_{\mathcal{G}}^+ := \{\mathbf{x} \in \mathcal{X} | x_1 = 0 \wedge x_2 > 0\}$ ; a touchdown set,  $\mathcal{X}_{\mathcal{G}}^- := \{\mathbf{x} \in \mathcal{X} | x_1 = 0 \wedge x_2 < 0\}$ ; and a ground crash set,  $\mathcal{X}_{\mathcal{G}}^0 := \{\mathbf{x} \in \mathcal{X} | x_1 = -1\}$ . Figure 2.2 illustrates this partition of the mechanical physical state space,  $\mathcal{X}$ .

In the stance set,  $\mathcal{X}_{\mathcal{G}} := \{\mathbf{x} \in \mathcal{X} | x_1 < 0\}$ , the mechanical dynamics is governed by the mass-spring-damper system in the absence of gravity,

$$\dot{\mathbf{x}} = \tilde{\mathbf{f}}_{\mathcal{G}}(\mathbf{x}, \xi) := \begin{bmatrix} 0 & 1 \\ -\xi^2 & -2\mu \end{bmatrix} \mathbf{x}, \quad \mathbf{x} \in \mathcal{X}_{\mathcal{G}}. \quad (2.3)$$

In the aerial set,  $\mathcal{X}_{\mathcal{A}} := \{\mathbf{x} \in \mathcal{X} | x_1 > 0\}$ , the body moves under the influence of unity gravitational acceleration,

$$\dot{\mathbf{x}} = \tilde{\mathbf{f}}_{\mathcal{A}}(\mathbf{x}, \xi) = \tilde{\mathbf{f}}_{\mathcal{A}}(\mathbf{x}) := \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ -1 \end{bmatrix}, \quad \mathbf{x} \in \mathcal{X}_{\mathcal{A}}. \quad (2.4)$$

Transitions between the aerial and stance modes are governed by the *lift-off* and the *touchdown* events. We will assume that both events are triggered when the body height,  $x_1$ , crosses the spring rest-length,  $\bar{x}_1 = 0$ . The hopper transitions from the aerial mode,  $\mathcal{A}$ , to the ground mode,  $\mathcal{G}$ , on the touchdown set,  $\mathcal{X}_{\mathcal{G}}^-$ , according to a plastic collision map,  $\tilde{\mathbf{c}} : \mathcal{X}_{\mathcal{A}} \rightarrow \mathcal{X}_{\mathcal{G}}$ ,

$$\tilde{\mathbf{c}}(\mathbf{x}) := \begin{bmatrix} x_1 \\ \zeta x_2 \end{bmatrix},$$

which is parametrized by the collision restitution coefficient,  $\zeta \in (0, 1]$ . The hopper transitions from the stance mode,  $\mathcal{G}$ , to the aerial mode,  $\mathcal{A}$ , on the touchdown set,  $\mathcal{X}_{\mathcal{G}}^+$ , where the mechanical states,  $\mathbf{x}$ , remain unchanged. The collision with the ground occurs on the ground crash set,  $\mathcal{X}_{\mathcal{G}}^0$ , where the body is assumed to stay stuck.

The two mechanical dissipation mechanisms, leg damping and plastic ground collision, are related. We distinguish three physical problem settings: 1) *lossy* where  $\mu > 0$ ; 2) *lossless* where  $\mu = 0$ ; and 3) *gainy* where  $\mu < 0$ .

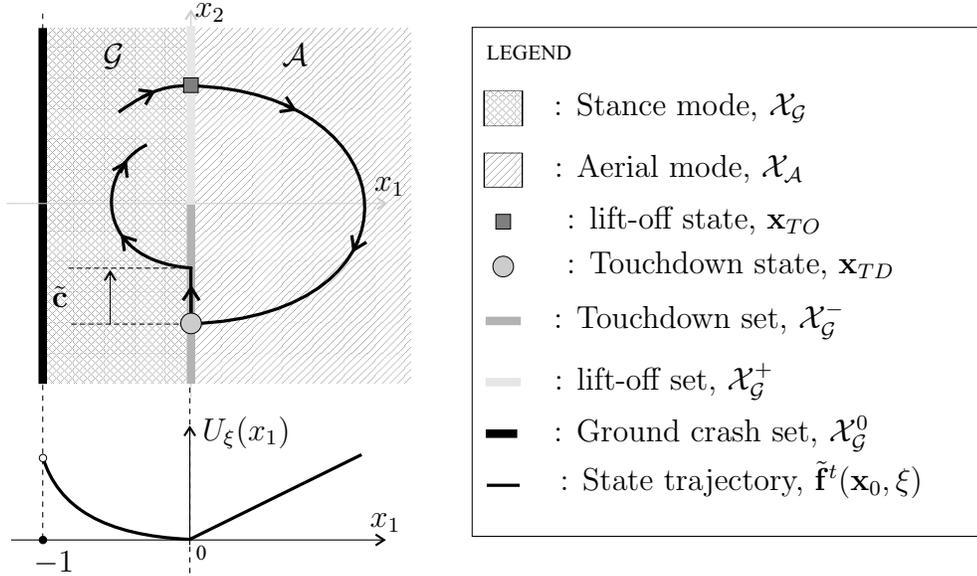


Figure 2.2: The top sketch depicts a typical state trajectory of the mechanical hopper,  $\tilde{\mathbf{f}}^t(\mathbf{x}_0, \xi)$ , with constant stiffness,  $\xi$ , where the four cells in the partition of the physical state space,  $\mathcal{X}$ , are listed in the accompanying legend. Plastic collision at touchdown,  $\mathbf{x}_{TD}$ , results in a reduction in total mechanical energy defined by the plastic collision map,  $\tilde{\mathbf{c}}(\mathbf{x}_{TD})$ . The bottom plot is the hybrid potential of the mechanical hopper,  $U_\xi(x_1)$ , composed of gravitational and Hooke's law spring potentials.

The mechanical potential energy of the hopper takes the form of a hybrid potential — a combination of the Hooke's law spring potential and the gravitational potential — given by

$$U_\xi(x_1) := \begin{cases} x_1 & ; x_1 \in (0, \infty) \\ 0 & ; x_1 = 0 \\ \frac{1}{2}\xi^2 x_1^2 & ; x_1 \in (-1, 0) \\ 0 & ; x_1 = -1 \end{cases}$$

and is illustrated in Figure 2.2 (bottom) along with a typical solution of (2.2) for constant leg stiffness,  $\xi$  (top). The resulting total mechanical energy is

$$H_\xi(\mathbf{x}) := \frac{1}{2}x_2^2 + U_\xi(x_1).$$

### 2.1.2 A Clock Controller

We posit a controller (as depicted in Figure 2.3) in the form of a one dimensional tunable “clock,” a dynamical system defined on the circle,  $S^1 := [0, 2\pi]/\{0, 2\pi\}$ ,

$$\dot{\theta} = 2\pi/T_c, \quad \theta \in \Theta := S^1, \quad (2.5)$$

where  $T_c \in \mathbb{R}^+$  is the clock period. A “shaping function,”  $\psi : \Theta \rightarrow \mathcal{K}$ , maps the clock phase,  $\theta$ , to the controller output. Hence, the output of the clock controller is a periodic signal whose detailed shape is dictated by the shaping function.

For the purposes of this analysis it will prove convenient to limit the shaping map,  $\psi(\theta)$ , to be an  $N$ -cell piecewise constant function,

$$\psi(\theta) = \xi_i, \quad \forall \theta \in \Theta_i := [\theta_i, \theta_{i+1}), \quad i \in \{1, 2, \dots, N\},$$

whose cells,  $\Theta_i$ , are defined by a monotonic discrete sequence of discontinuity points,  $\{\theta_1, \theta_2, \dots, \theta_{N+1}\} \subset \Theta$ , where the smallest difference between any two consecutive entries is bounded from below,  $\theta_{i+1} - \theta_i > \delta\theta > 0$  for all  $i = 1, \dots, N$ . Without any loss of generality we will assume that  $\theta_1 = \theta_{N+1} = 0$ .

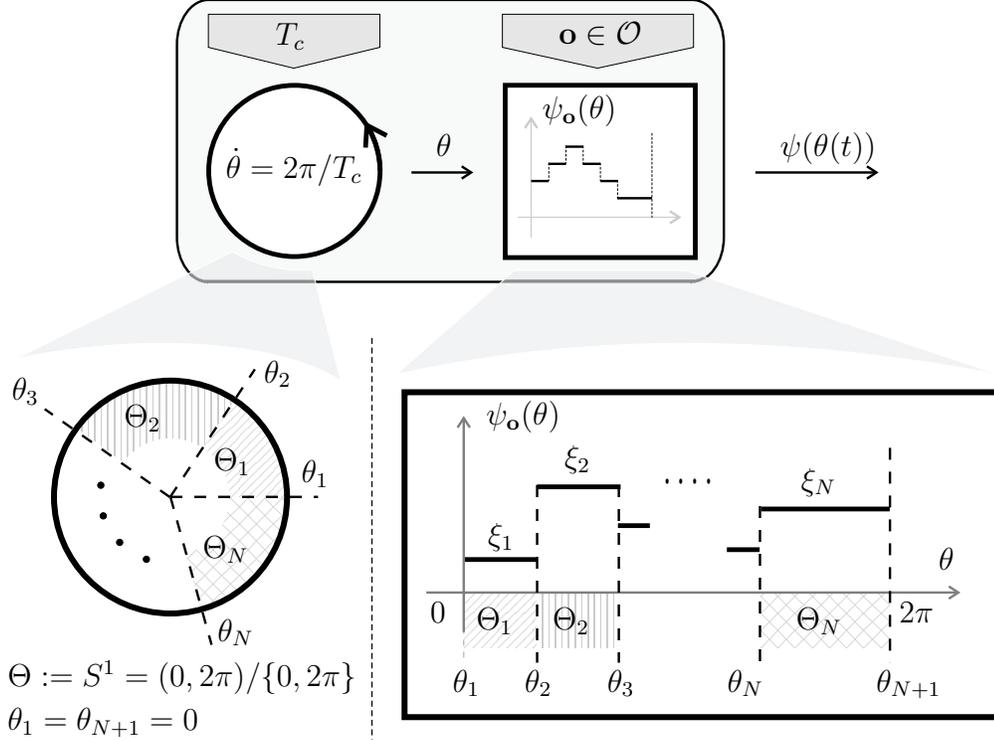


Figure 2.3: An illustration of a clock controller. It consists of a clock and an output function,  $\psi(\theta)$ , which is an  $N$ -cell piecewise constant function mapping the clock phase to the scalar output. The output function,  $\psi(\theta)$ , is parameterized by the shape configuration,  $\mathbf{o} \in \mathcal{O}$ . The controller output,  $\psi(\theta(t))$ , is a piece-wise constant periodic signal.

The resulting controller output,  $\psi(\theta(t))$ , is a piecewise constant signal with period  $T_c$  and is parameterized by the “shape vector,”  $\mathbf{o} \in \mathcal{O} \subset S^{(N-1)} \times \mathbb{R}^N$ , where the “shape configuration space,”  $\mathcal{O}$ , is defined as

$$\mathcal{O} := \{[\theta_2, \dots, \theta_N]^T \in \Theta^{N-1} \mid \theta_{i+1} - \theta_i > \delta\theta > 0\} \times \mathcal{K}^N.$$

### 2.1.3 A Unidirectional Coupling Scheme

To construct the *clock driven 1-DOF hopper*, the central object of our study, we couple the clock controller (2.5) with the the mechanical hopper (2.2) in a feed-forward fashion such that the controller output,  $\psi$ , modulates the leg stiffness,  $\xi$ , as stated by

$$\xi(t) := \psi(\theta(t)).$$

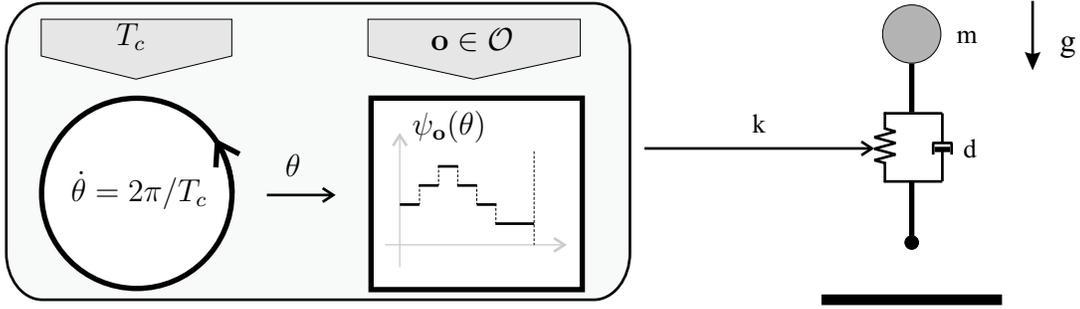


Figure 2.4: Clock driven 1-DOF hopper setting. The mechanical hopper and the clock controller are forward coupled such that the periodic output of the controller,  $\psi(\theta(t))$ , modulates the leg stiffness,  $\xi$ . The clock controller is parameterized by the clock period,  $T_c$ , and the shape configuration,  $\mathbf{o}$ , which are considered adjustable.

This one-way coupling of (2.2) and (2.5) yields a three dimensional autonomous nonlinear dynamical system,

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{f}}(\mathbf{x}, \psi(\theta)) \\ 2\pi/T_c \end{bmatrix}, \quad \begin{bmatrix} x_1 \\ x_2 \\ \theta \end{bmatrix} \in \mathcal{X} \times \Theta, \quad (2.6)$$

whose hybrid structure is defined by the cross product of the modes of the mechanical hopper,  $\{\mathcal{G}, \mathcal{A}\}$ , and the cells of the clock controller,  $\{\Theta_i\}_{i=1}^N$ . The resulting (coupled)

hybrid system has  $2N$  modes consisting of  $N$  aerial modes,  $\mathcal{A}_i := \mathcal{A} \times \Theta_i$ ,  $i = 1, 2, \dots, N$ , with identical dynamics,

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{f}}_{\mathcal{A}}(\mathbf{x}) \\ 2\pi/T_c \end{bmatrix}, \quad \begin{bmatrix} \mathbf{x} \\ \theta \end{bmatrix} \in \mathcal{A}_i; \quad (2.7)$$

and;  $N$  stance modes,  $\mathcal{G}_i := \mathcal{G} \times \Theta_i$ ,  $i = 1, 2, \dots, N$ , where dynamics of each mode,

$$\begin{bmatrix} \dot{\mathbf{x}} \\ \dot{\theta} \end{bmatrix} = \begin{bmatrix} \tilde{\mathbf{f}}_{\mathcal{G}}(\mathbf{x}, \xi_i) \\ 2\pi/T_c \end{bmatrix}, \quad \begin{bmatrix} \mathbf{x} \\ \theta \end{bmatrix} \in \mathcal{G}_i, \quad (2.8)$$

is parametrized by the constant value of the controller output,  $\xi_i$ , during its associated clock cell,  $\Theta_i$ . We introduce a shorthand notation for the stance mode vector field during the  $i^{\text{th}}$  clock cell,  $\Theta_i$ ,

$$\tilde{\mathbf{f}}_i(\mathbf{x}) \equiv \tilde{\mathbf{f}}_{\mathcal{G}}(\mathbf{x}, \xi_i),$$

which will be employed hereafter.

## 2.2 Groundwork for the Analysis

### 2.2.1 Preferred Coordinates and Related Transformations

In the physical coordinate system,  $\mathcal{X}$ , the flow of the mechanical hopper states in all aerial modes,  $\mathcal{A}_i$ ,  $i = 1, 2, \dots, N$ ,

$$\tilde{\mathbf{f}}_{\mathcal{A}}^t(\mathbf{x}(t_0), \xi) = \tilde{\mathbf{f}}_{\mathcal{A}}^t(\mathbf{x}(t_0)) = \begin{bmatrix} -\frac{1}{2}(t-t_0)^2 + x_2(t_0)(t-t_0) + x_1(t_0) \\ -(t-t_0) + x_2(t_0) \end{bmatrix}, \quad (2.9)$$

is simple enough that there is no virtue in changing coordinates.

On the other hand, we prefer to study the hopper dynamics during each stance mode,  $\mathcal{G}_i$ , in an associated action-angle coordinate system,  $\mathcal{E}_i := \mathbb{R}^+ \times S^1$  — the polar coordinate representation of the Real Jordan Canonical form of the physical state space,  $\mathcal{X}$  — defined by the transformation,  $\mathbf{g}_i : \mathcal{X} \rightarrow \mathcal{E}_i$ ,

$$\mathbf{e} = \begin{bmatrix} \eta \\ \phi \end{bmatrix} = \mathbf{g}_i(\mathbf{x}) = \begin{bmatrix} \|\mathbf{y}\|_2 \\ \arctan\left(\frac{-y_1}{-y_2}\right) \end{bmatrix}, \quad \text{where } \mathbf{y} = \begin{bmatrix} \xi_i & \mu/\xi_i \\ 0 & \gamma_i/\xi_i \end{bmatrix} \mathbf{x}. \quad (2.10)$$

Hereafter, we will refer to this family of coordinate systems as “energy-phase” coordinates. Note that (2.10) is parametrized by the constant leg stiffness,  $\xi = \xi_i$ , chosen during the  $i^{\text{th}}$  clock cell,  $\theta \in \Theta_i$ , and therefore, the energy-phase coordinate system associated with a particular stance phase is distinct from the rest. The states of the energy-phase coordinates are the normalized mechanical energy,  $\eta \in \mathbb{R}^+$ , and the normalized mechanical phase,  $\phi \in S^1$ . Here, we identify the natural mechanical oscillation frequency during the  $i^{\text{th}}$  clock cell,  $\Theta_i$ ,

$$\gamma_i := \sqrt{\xi_i^2 - \mu^2},$$

which is real and positive definite,  $\gamma_i \in \mathbb{R}^+$ , for all admissible normalized stiffness values,  $\xi_i \in \mathcal{K} := (\mu, \infty)$ . In the  $i^{\text{th}}$  energy-phase space,  $\mathcal{E}_i$ , the stance and aerial partitions are separated by the  $i^{\text{th}}$  touchdown and lift-off phases given by

$$\phi_{TD}^i := \arctan(\mu/\gamma_i), \quad \text{and} \quad \phi_{LO}^i := \phi_{TD}^i + \pi, \quad (2.11)$$

respectively.

In this preferred coordinate system the stance dynamics is given by

$$\dot{\mathbf{e}} = \mathbf{f}_i(\mathbf{e}) = \begin{bmatrix} -\mu\eta \\ \gamma_i \end{bmatrix}$$

whose flow,

$$\mathbf{f}_i^t(\mathbf{e}(t_0)) = \begin{bmatrix} \eta(t_0)\exp[-\mu(t-t_0)] \\ \phi(t_0) + \gamma_i(t-t_0) \end{bmatrix}, \quad (2.12)$$

is conjugate on  $\mathcal{X}$  to that of (2.3) via  $\mathbf{f}_i^t \equiv \mathbf{g}_i \circ \tilde{\mathbf{f}}_{\mathcal{G}}^t \circ \mathbf{g}_i^{-1}$  when  $\xi = \xi_i$ .

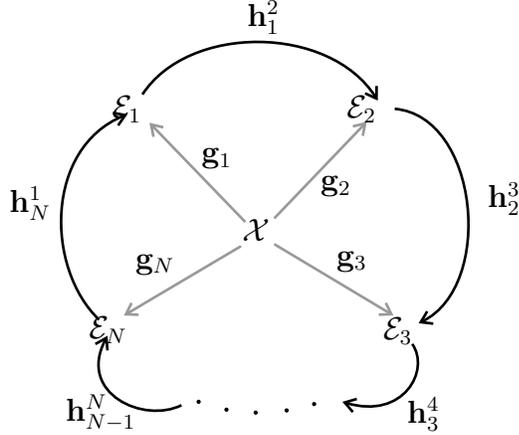


Figure 2.5: Coordinate systems and the transformations that relate them.

The parametric dependence of the transformation,  $\mathbf{g}_i$ , on the stiffness,  $\xi = \xi_i$ , causes the energy-phase coordinate systems for each cell,  $\mathcal{E}_i$ , to be different as depicted in the commutative diagram of Figure 2.5. The energy-phase coordinate systems of  $i^{\text{th}}$  and  $j^{\text{th}}$  cells are related by the “relating transformation,”  $\mathbf{h}_i^j : \mathcal{E}_i \rightarrow \mathcal{E}_j$ ,

$$\mathbf{h}_i^j(\mathbf{e}) := \mathbf{g}_j \circ \mathbf{g}_i^{-1}(\mathbf{e}) = \begin{bmatrix} \eta \cdot \mathbf{a}_i^j(\phi) \\ \mathbf{b}_i^j(\phi) \end{bmatrix}, \quad (2.13)$$

whose energy component<sup>2</sup>,  $\pi_1 \circ \mathbf{h}_i^j(\mathbf{e})$ , is linear in energy,  $\eta$ , and scaled by the “energy map coefficient,”

---

<sup>2</sup>Henceforth,  $\pi_i \circ \mathbf{f} \equiv (\mathbf{f})_i$  will refer to the projection of any function,  $\mathbf{f} : \mathbb{R}^n \rightarrow \mathbb{R}^n$  onto the  $i^{\text{th}}$  coordinate.

$$\mathbf{a}_i^j(\phi) := \frac{\gamma_j \xi_i}{\gamma_i \xi_j} |\cos(\phi)| \sqrt{1 + [\mathbf{H}_i^j \circ \tan(\phi)]^2}, \quad (2.14)$$

while its phase component,  $\pi_2 \circ \mathbf{h}_i^j(\mathbf{e})$ , is a monotonic map on the circle,  $\mathbf{b}_i^j : S^1 \rightarrow S^1$ ,

$$\mathbf{b}_i^j(\phi) := \arctan \circ \mathbf{H}_i^j \circ \tan(\phi), \quad (2.15)$$

where

$$\mathbf{H}_i^j(s) := \frac{\gamma_i \xi_j^2}{\gamma_j \xi_i^2} s + \frac{\mu}{\gamma_j} \left[ 1 - \left( \frac{\xi_j}{\xi_i} \right)^2 \right]. \quad (2.16)$$

In a relating transformation expression,  $\mathbf{h}_i^j$ , the subscript will be referred to as its “source index” and the superscript as its “target index.” Two relating transformations are called “consecutive” if the source index of one is the same with the target index of the other.

Derivation of the return map in Section 2.4.1 and its analysis in the subsequent sections will heavily utilize the two basic properties of the relating transformations: 1) *inversion property*,  $(\mathbf{h}_i^j)^{-1} \equiv \mathbf{h}_j^i$ ; and 2) *consecutive composition property*,  $\mathbf{h}_k^j \circ \mathbf{h}_i^k \equiv \mathbf{h}_i^j$ . Note also that  $(\mathbf{H}_i^j)^{-1} \equiv \mathbf{H}_j^i$ . Finally, Lemma 1 states a special property of the phase component of the relating transformation.

**Lemma 1.** *The touchdown and lift-off phases of two energy-phase coordinate systems are related by the phase component of the transformation that relates these two spaces,*

$$\phi_{TD}^j = \mathbf{b}_i^j(\phi_{TD}^i) \quad \text{and} \quad \phi_{LO}^j = \mathbf{b}_i^j(\phi_{LO}^i) \quad (2.17)$$

PROOF. The touchdown,  $\phi_{TD}^i$ , and lift-off phases,  $\phi_{LO}^i$ , are defined in (2.11). Result follows from the evaluation of the phase component of the relating transformation,

$b_i^j(\phi)$ , at these special phases by direct computation. □

## 2.2.2 Basic Setup of the Poincaré Analysis

We will study the limit behavior of the coupled system in (2.6) using the *Poincaré method* — a powerful tool for the analysis of cyclic limit behaviors [120]. This section will introduce the basic setup for this analysis.

As a convention we will study the continuous-time dynamics of the mechanical system (hopper) in the energy-phase space of the  $N^{\text{th}}$  stance mode,  $\mathcal{E}_N$ . In this preferred coupled state space,  $\mathcal{E}_N \times \Theta$ , we define a 2-dimensional sub-manifold,

$$\Sigma := \{\mathcal{E}_N \times \Theta \mid \theta = 0\}, \quad (2.18)$$

by imposing a condition on the clock phase. By the virtue of clock dynamics in (2.5) all flows of the coupled system in (2.6) punctures through  $\Sigma$  and all state trajectories originating from  $\Sigma$  will return back to it after a finite time period (in fact, exactly after one clock period,  $T_c$ ). Hence,  $\Sigma$  is a well-defined Poincaré section.

Sampling the coupled states,  $[\eta, \phi, \theta]^T$ , on the event of return to  $\Sigma$  relates the three dimensional continuous-time system (2.6) to its corresponding two dimensional discrete-time Poincaré map,

$$\mathbf{p}_{i+1} = \mathbf{r}(\mathbf{p}_i), \quad \mathbf{p} = \begin{bmatrix} \nu \\ \alpha \end{bmatrix} \in \mathcal{P} := \mathcal{V} \times \mathcal{D}. \quad (2.19)$$

where  $\mathcal{V} := \mathbb{R}^+$  and  $\mathcal{D} := [0, 2\pi/\gamma_N)$ .

The default state space for the discrete-time Poincaré map is defined by the independent states in the Poincaré section,  $\Sigma$ , which are the normalized mechanical energy,  $\eta \in \mathbb{R}^+$ , and the normalized mechanical phase,  $\phi \in S^1$ , in the preferred me-

chanical coordinates,  $\mathcal{E}_N$ . However, we choose to work in an alternative<sup>3</sup> “Poincaré space,”  $\mathcal{P} := \mathcal{V} \times \mathcal{D} \subset \mathbb{R}^+ \times S^1$ , defined by  $\mathbf{n} : \mathcal{E}_N \rightarrow \mathcal{P}$ ,

$$\mathbf{n}^{-1}(\mathbf{p}) = \begin{bmatrix} \nu \cdot \zeta \cdot \exp[-\mu\alpha] \\ \mathsf{P}(\alpha) \end{bmatrix}, \quad (2.20)$$

where

$$\mathsf{P}(\alpha) := \gamma_N \alpha + \phi_{TD}^N.$$

For those operating regimes where the clock cycle concludes in the stance mode,  $\mathbf{x}(\theta = 2\pi) \in \mathcal{X}_G$ , and the last hopper touchdown event occurs during the  $N^{\text{th}}$  clock cell,  $\max\{\theta \mid \mathbf{x} \in \partial\mathcal{X}_G^-\} \in \Theta_N$ , the Poincaré states,  $\nu \in \mathcal{V}$  and  $\alpha \in \mathcal{D}$ , have physical interpretations. The former,  $\nu \in \mathcal{V}$ , is the normalized speed at the last touchdown (equivalently, total mechanical energy at touchdown), and will be referred as the “speed” state. The latter,  $\alpha \in \mathcal{D}$ , is the time delay between the last touchdown and the clock reset,  $\theta = 0$ , and will be referred as the “delay” state. In fact, the delay state,  $\alpha$ , is a surrogate for the phase difference between the controller clock and the mechanical clock (hopper).

## 2.3 Mode Sequences

### 2.3.1 Definition and Significance

Computation of a global return map,  $\mathbf{r}(\mathbf{p})$ , which captures all possible transitions between two Poincaré samples is prohibitively difficult if not effectively impossible. Instead we choose to partition the problem into manageable parts and investigate them separately.

---

<sup>3</sup>This coordinate system was proposed by Prof. Philip Holmes in a personal communication [127]

At the heart of this divide-and-conquer approach lies the observation that each returned state,  $\mathbf{p} \in \mathcal{P}$ , can be assigned a unique “mode sequence,”  $\sigma$  — a string over the alphabet of coupled system modes,  $\{\mathcal{G}_i, \mathcal{A}_i\}_{i=1}^N$  — according to the sequence of open-cell modes (we strip off the boundary mode names since they contribute no information concerning the flow through the open cells,  $\mathcal{X}_G^-, \mathcal{X}_G^+$  and  $\mathcal{X}_G^0$ ) that its future continuous-time trajectory passes through on the way to the next return.

We partition the Poincaré space,  $\mathcal{P}$ , by mode sequence and label each cell of this partition by its (common) mode string,  $\sigma$ . Figure 2.6 illustrates a numerically computed partition for a coupled system with 2-cell output function.

For a given mode sequence,  $\sigma$ , one can easily compute the return map,  $\mathbf{r}(\mathbf{p})$ , by appropriately composing the flow through each mode in (2.8) and (2.7) visited by the coupled system. This allows us to conduct local stability analysis within each cell and will be the basis of the analysis that will be presented in Section 2.4. Henceforth, the return map that is computed for a particular operating regime characterized by a mode sequence,  $\sigma$ , will be referred as “the return map of the mode sequence,  $\sigma$ ” and will be denoted by  $\mathbf{r}_\sigma(\mathbf{p})$ .

Naturally, a return map,  $\mathbf{r}(\mathbf{p})$ , of a particular mode sequence,  $\sigma$ , is only *valid* within the confines of the cell that its mode sequence defines. We will refer to this subset of the Poincaré space as “the domain of the mode sequence” and denote it by  $\mathcal{P}(\sigma)$ .

### 2.3.2 Classes of Mode Sequences

This section will introduce terminology to aid the discussions concerning mode sequences. Note that *only* those mode sequences that can be generated by the coupled system in (2.6) have any physical relevance. The collection of these mode sequences, which will be referred as the “set of realizable mode sequences” and denoted by  $\mathcal{M}$ ,

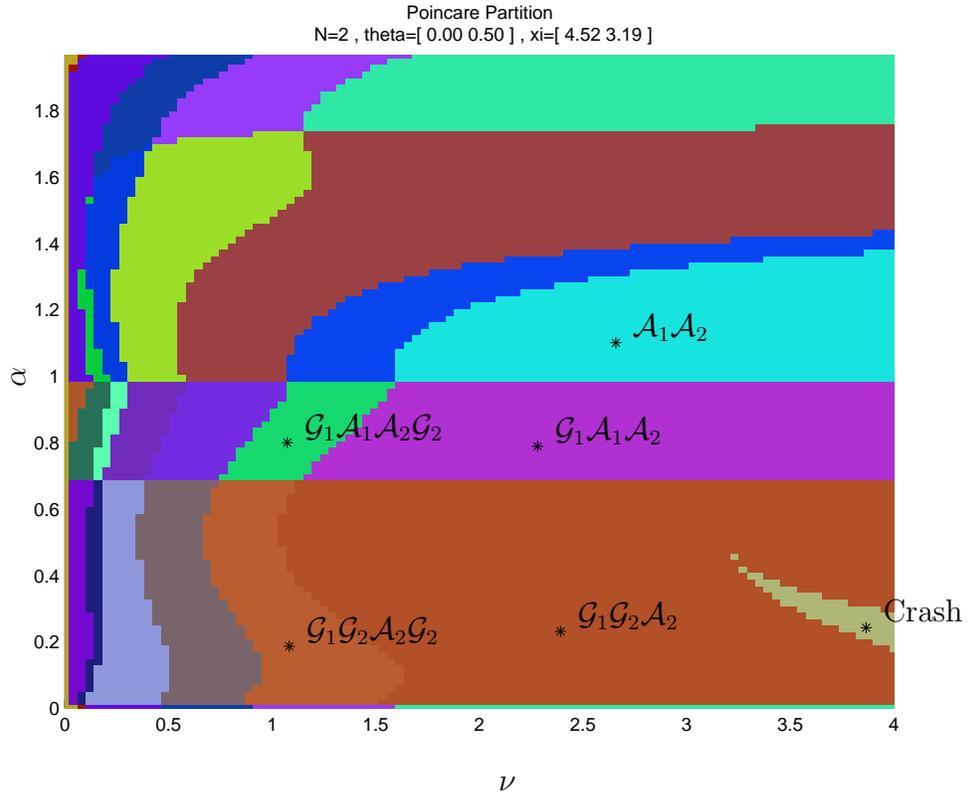


Figure 2.6: A numerically computed partition of the Poincaré state space,  $\mathcal{P}$ , according to the common mode sequences,  $\sigma$ , for a coupled system with 2-cell output function whose parameters are given in the title. Some of the mode domains are pointed out along with the set of Poincaré states that lead to ground crash.

defines a language over the alphabet of modes,

$$\mathcal{M}(\mu, \zeta, \mathbf{o}, T_c) \subset \left( \{\mathcal{G}_i, \mathcal{A}_i\}_{i=1}^N \right)^*$$

parametrized by the viscous damping,  $\mu$ ; plastic collision coefficient,  $\zeta$ ; controller shape vector,  $\mathbf{o} \in \mathcal{O}$ ; and clock period,  $T_c$ . For notational simplicity, hereafter, we will omit these parameters when we refer to this set. Furthermore, realizability of mode sequences will not be explicitly stated in the following sections but the reader must assume that all discussions concern realizable mode sequences.

Driven by the dynamics of the coupled system in (2.6) the realizable mode sequences,  $\sigma \in \mathcal{M}$ , adopt a particular grammar characterized by two rules: 1) mode indices start from 1 and increment monotonically, completely spanning the set  $\{1, 2, \dots, N\}$ ; and 2) for those substrings where indices remain the same the two mechanical modes,  $\{\mathcal{G}, \mathcal{A}\}$ , alternate.

This structure inspires two shorthand notations: 1) a subscript  $[i, j]$  to a mechanical mode denotes a sequence of basic modes of the referred mechanical mode with indices strictly monotonically increasing from  $i$  to  $j$ ,  $\mathcal{G}_{[i,j]} \equiv \mathcal{G}_i \mathcal{G}_{i+1} \dots \mathcal{G}_{j-1} \mathcal{G}_j$ , and; 2) a superscript  $n$  to the parenthesis around a substring denotes that the referred substring repeats  $n$  times,  $(\mathcal{G}_i \mathcal{A}_i)^n \equiv \mathcal{G}_i \mathcal{A}_i \mathcal{G}_i \mathcal{A}_i \dots \mathcal{G}_i \mathcal{A}_i$ . We will heavily employ these shorthand notations in the following sections for notational ease.

We believe that the structure of a mode sequence string,  $\sigma$ , hints at the dynamic characteristics of the associated operating regime of the coupled system in (2.6). This section will define several interesting classes of the (realizable) mode sequences and discuss their stability properties. These observations will guide our work in the later sections toward interesting and relevant operating conditions.

A mode sequence,  $\sigma$ , is said to be “stable” if its associated return map,  $\mathbf{r}(\mathbf{p})$ , has a stable fixed point,  $\mathbf{p}^*$ , in its domain,  $\mathcal{P}(\sigma)$ . Conversely, a mode sequence is “unstable” if its domain does not contain any stable fixed point.

We say that a mode sequence is “repeatable” if it satisfies two conditions: 1) the sequence starts and ends with the same mechanical mode; and 2) the sequence contains each mechanical mode,  $\{\mathcal{G}, \mathcal{A}\}$ , at least once, that is, it possesses both a stance and a flight phase. Naturally, all the mode sequences that fail to satisfy these conditions will be referred as “non-repeatable.” According to the repeatability property we partition the mode sequence set,  $\mathcal{M}$ , into two: 1) the set of repeatable mode sequences,  $\mathcal{M}_r$ ; and 2) the set of non-repeatable mode sequences,  $\mathcal{M}_n := \overline{\mathcal{M}_r}$ .

The mode sequence,  $\sigma$ , implicitly specifies the sequence of mechanical transitions, touchdown and lift-off, observed from one Poincaré sample to the next. According to this sequence of mechanical transitions we partition the set of mode sequences,  $\mathcal{M}$ , into three groups: “fundamentals,”  $\mathcal{M}^0$ , containing a single touchdown and lift-off event; “sub-harmonics,”  $\mathcal{M}_-$ , containing either no transitions or just a single one; and, “super-harmonics,”  $\mathcal{M}_+$ , containing multiple touchdown and lift-off transitions.

In this study we will limit our attention on fundamental mode sequences,  $\mathcal{M}^0$ , where the controller clock and the mechanical hopper become phase locked, and therefore, share the same period, at a stable limit behavior. Our studies suggest that within the set of fundamental mode sequences,  $\mathcal{M}^0$ , only those that are repeatable may have a stable fixed point in their corresponding domain, and the rest are unstable. Hence, our discussions in the remainder of this thesis will specifically consider *fundamental repeatable mode sequences*. Note that this set of behaviors capture all period-one hopping gaits, and therefore, constitutes a physically relevant and interesting set of operating regimes.

## 2.4 Fundamental Repeatable Mode Sequences

In the analysis of the coupled system (2.6) we will only consider the repeatable mode sequences. The set of repeatable mode sequences,  $\mathcal{M}_r$ , contain both super-harmonic and fundamental mode sequences. This section will focus on the latter due to their relatively simple mode sequence structures. The reader should note that the method can also facilitate the analysis of the former as well.

In particular, we will investigate a family of mode sequences characterized by

$$\mathcal{GAG}(e, N) := \mathcal{G}_{[1,e]} \mathcal{A}_{[e,N]} \mathcal{G}_N$$

where  $N > 1$  and  $e \in \{1, 2, \dots, N\}$  is the clock cell when the lift-off occurs. Note that any fundamental repeatable mode sequence can be converted into being a member of this particular family by appropriately renumbering the clock cells. Therefore, the results of the following discussions apply to the entire fundamental repeatable mode sequence family.

In Section 2.4.1 we will first explain in detail how to compute the return map,  $\mathbf{r}(\mathbf{p})$ , for a member of this family,  $\sigma \in \mathcal{GAG}(e, N)$ . The same section also analytically characterizes the domain of the mode sequence,  $\mathcal{P}(\sigma)$ . Next, we identify physically viable (valid) fixed points,  $\mathbf{p}^*$ , of the return map,  $\mathbf{r}(\mathbf{p})$ , and present a set of sufficient conditions leading to their local stability in Section 2.4.4.

### 2.4.1 Return Map Derivation

For fundamental repeatable mode sequences,  $\mathcal{GAG}(e, N)$ , the preferred Poincaré states in Section 2.2.2 have physical interpretations: the speed state,  $\nu \in \mathcal{V}$ , is the normalized total mechanical energy at the touchdown; and, the delay state,  $\alpha \in \mathcal{D}$ , is the time delay between touchdown and clock cycle end. This physical insight will be frequently employed in the following explanations. Figure 2.7 depicts a typical coupled system state trajectory projected onto the mechanical state space,  $\mathcal{X}$ , and will be used as a visual aid for the return map derivations.

For notational ease we define a “chain composition” operator<sup>4</sup>,

$$\begin{array}{c} j \\ \odot \quad [\mathbf{F}_k] := \mathbf{F}_j \circ \mathbf{F}_{j-1} \circ \dots \circ \mathbf{F}_{i+1} \circ \mathbf{F}_i, \\ k = i \end{array}$$

---

<sup>4</sup>In the following derivations a key function in (2.23) is defined as a composition of a list of maps. Using this new symbol we aim to avoid any confusion that may arise when we compute derivatives in (2.46).

denoting an ordered composition of a family of functions,  $\mathbf{F}_i$ .

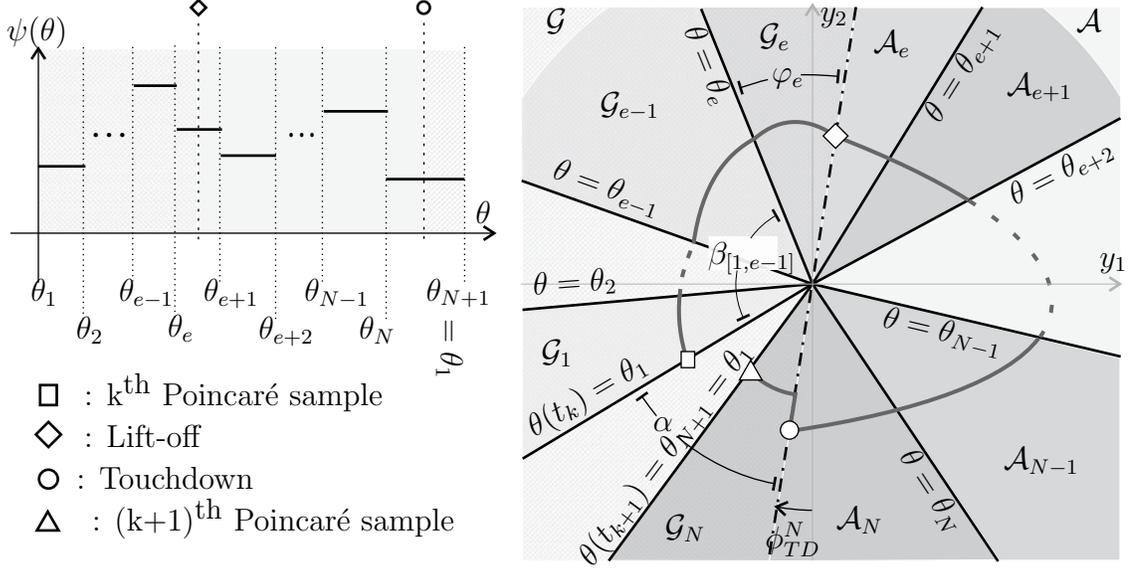


Figure 2.7: A typical output function with  $N$ -cells (left) and a typical hopper state trajectory in the physical coordinate system,  $\mathcal{X}$ , between two consecutive Poincaré samples (right). Crucial events along the trajectory are indicated by markers whose list can be found in the legend below.

By appropriately composing the mechanical flows through the stance,  $\mathbf{G}_i^t := \mathbf{h}_i^N \circ \mathbf{f}_i^t \circ \mathbf{h}_N^i$ , and the aerial modes,  $\mathbf{A}_i^t := \mathbf{g}_N \circ \tilde{\mathbf{f}}_{\mathcal{A}}^t \circ \mathbf{g}_N^{-1}$ , as well as the touchdown collision map,  $\mathbf{c} := \mathbf{g}_N \circ \tilde{\mathbf{c}} \circ \mathbf{g}_N^{-1}$ , we compute the return map,  $\mathbf{r} : \mathcal{P} \rightarrow \mathcal{P}$ ,

$$\mathbf{r} := \mathbf{n} \circ \mathbf{G}_N^\alpha \circ \mathbf{c} \circ \mathbf{A}_N^{(\beta_N - \alpha)} \circ \begin{pmatrix} N-1 \\ \odot \\ j=e+1 \end{pmatrix} \circ [\mathbf{A}_j^{\beta_j}] \circ \mathbf{A}_e^{(\beta_e - \varphi_e)} \circ \mathbf{G}_e^{\varphi_e} \circ \begin{pmatrix} e-1 \\ \odot \\ j=1 \end{pmatrix} \circ \mathbf{n}^{-1},$$

relating the  $k^{\text{th}}$  Poincaré sample,  $\mathbf{p}_k$ , to the next,  $\mathbf{p}_{k+1}$ . In this formula

$$\beta_j := (\theta_{j+1} - \theta_j) / T_c \quad (2.21)$$

represents the constant duration of the  $j^{\text{th}}$  clock cell. We define “remaining time” as the time spent in ground contact during the  $e^{\text{th}}$  cell prior to lift-off and denote by  $\varphi_e$ .

**Proposition 1.** *The return map of the clock controlled linear hopper, restricted to the mode sequence  $\mathcal{GAG}(e, N)$ , has the structure*

$$\mathbf{p}_{k+1} = \mathbf{r}(\mathbf{p}_k); \quad \mathbf{r}(\nu, \alpha) := \begin{bmatrix} \nu\rho(\alpha) \\ T_c - \beta_{[1, e-1]} - \varphi_e(\alpha) - 2\nu\rho(\alpha) \end{bmatrix}, \quad (2.22)$$

where  $\rho(\alpha)$  is the forcing function (2.26),  $\varphi_e(\alpha)$  is the remaining time function (2.29), and  $\beta_{[1, e-1]}$  is the total duration of the cells completely swept during ground phase (2.28).

PROOF. We will compute the return map,  $\mathbf{r}(\mathbf{p})$ , in two parts: the energy component,  $\mathbf{r}_1(\mathbf{p}) := \pi_1 \circ \mathbf{r}(\mathbf{p})$ ; and the delay component,  $\mathbf{r}_2(\mathbf{p}) := \pi_2 \circ \mathbf{r}(\mathbf{p})$ . Recall that the Poincaré states correspond to normalized physical quantities, namely, the speed state,  $\nu$ , is the normalized total mechanical energy at touchdown and the delay state,  $\alpha$ , is the normalized time between onset of the clock period and the last touchdown.

First we start with the derivation of the energy component of the return map,  $\mathbf{r}_1(\mathbf{p})$ , where we employ our insight in the mechanical system to simplify its expression. Recall that the aerial hopper dynamics given in (2.4) are governed by the conservative gravitational potential field, and the lift-off,  $\partial\mathcal{X}_G^+$  and touchdown,  $\partial\mathcal{X}_G^-$ , boundaries lie on the same potential level set. Hence, the aerial phase of this mode sequence,  $\mathcal{A}_{[e, N]}$ , causes no change in the total mechanical energy,

$$\pi_1 \circ \mathbf{A}_N^{(\beta_N - \alpha)} \circ \begin{pmatrix} N - 1 \\ \odot \quad [\mathbf{A}_j^{\beta_j}] \\ j = e + 1 \end{pmatrix} \circ \mathbf{A}_N^{(\beta_e - \varphi_e)}(\mathbf{e}) \equiv id_{\mathbb{R}^+}(\eta).$$

Upon touchdown, the plastic collision map,  $\mathbf{c}(\mathbf{e})$ , which maps the touchdown boundary,  $\mathbf{g}_N(\partial\mathcal{X}_G^-) = \{\mathcal{E}_N | \phi = \phi_{TD}^N\}$  onto itself, scales the total mechanical energy,  $\pi_1 \circ \mathbf{c}(\mathbf{e}) = \zeta \cdot \eta$ , while keeping the mechanical phase constant,  $\pi_2 \circ \mathbf{c}(\mathbf{e}) = \phi$ . Noting  $\pi_1 \circ \mathbf{n} \circ \mathbf{G}_N^\alpha \equiv id_{\mathbb{R}^+}(\eta)$  along the touchdown boundary,  $\mathbf{e} \in \mathbf{g}_N(\partial\mathcal{X}_G^-)$ , allows further simplification of the energy component of the return map,  $\mathbf{r}_1(\mathbf{p})$ . We expand the remaining expression employing the consecutive composition property of the relating transformations,  $\mathbf{h}_k^j \circ \mathbf{h}_i^k \equiv \mathbf{h}_i^j$ , and letting  $\mathbf{h}_0^1 \equiv \mathbf{h}_N^1$ ,

$$\mathbf{r}_1(\mathbf{p}) = \pi_1 \circ \mathbf{c} \circ \mathbf{h}_e^N \circ \mathbf{f}_e^{\varphi_e} \circ \mathbf{h}_{e-1}^e \circ \begin{pmatrix} e - 1 \\ \odot \quad [\mathbf{f}_j^{\beta_j} \circ \mathbf{h}_{j-1}^j] \\ j = 1 \end{pmatrix} \circ \mathbf{n}^{-1}(\mathbf{p}).$$

which, in turn, can be rewritten by exploiting the linearity of the terms in the energy state,  $\eta$ ,

$$\mathbf{r}_1(\mathbf{p}) = \zeta \cdot \mathbf{a}_e^N(\phi_{LO}^e) \cdot \exp[-\mu\varphi_e] \cdot \mathbf{a}_{e-1}^e(\phi_e^*) \cdot \left( \prod_{j=1}^{e-1} [\exp[-\mu\beta_j] \cdot \mathbf{a}_{j-1}^j(\phi_j^*)] \right) \cdot \exp[-\mu\alpha] \cdot \nu$$

where  $\phi_{LO}^e := \pi + \phi_{TD}^e = \pi + \arctan(\mu/\gamma_e)$  is the mechanical phase in  $\mathcal{E}_e$  coordinates at the lift-off event and  $\phi_i^* := \pi_2 \circ \mathbf{g}_{i-1}(\mathbf{x}(t_i))$  is the phase of the mechanical system in  $\mathcal{E}_{i-1}$  coordinates at the instant,  $t_i$ , when the clock switches from the  $(i-1)^{\text{th}}$  cell to the  $i^{\text{th}}$  cell,  $\theta(t_i) = \theta_i$ . Introducing a family of affine transformations,  $M_j(s) := s + \gamma_j\beta_j$  for  $j \in \{1, \dots, N\}$ , the switching phases,  $\phi_i^*$ , can be rewritten as

$$\phi_i^*(\alpha) := \left[ \begin{array}{c} i-1 \\ \odot \quad (M_j \circ b_{j-1}^j) \\ j=1 \end{array} \right] \circ P(\alpha). \quad (2.23)$$

The final simplification comes from the fact that lift-off occurs when the leg spring reaches its rest length and it can be shown by direct computation that  $a_e^N(\phi_{LO}^e) \equiv 1$  for all  $e \in \{1, \dots, N\}$ .

We find it useful to collect the multiplicative terms of the energy component,  $\mathbf{r}_1(\mathbf{p})$ , in two groups: a loss term

$$\epsilon(\alpha) := \zeta \cdot \exp \left[ -\mu(\alpha + \varphi_e(\alpha) + \beta_{[1,e-1]}) \right]; \quad (2.24)$$

and, a set of action terms (each capturing an instantaneous stiffness change during stance mode,  $\mathcal{G}$ ),

$$l_j(\alpha) := a_{j-1}^j \circ \phi_j^*(\alpha), \quad j \in \{1, 2, \dots, e\}; \quad (2.25)$$

leading to a multiplicative decomposition of the energy component of the return map,

$$\mathbf{r}_1(\mathbf{p}) := \nu \rho(\alpha), \quad \rho(\alpha) := \epsilon(\alpha) \left[ \prod_{j=1}^e l_j(\alpha) \right]. \quad (2.26)$$

Hereafter,  $\rho(\alpha)$  will be referred as the “forcing function.”

Second, we derive the delay component of the return map,  $\mathbf{r}_2(\mathbf{p}) := \pi_2 \circ \mathbf{r}(\mathbf{p})$ . Recall the chosen Poincaré section,  $\Sigma$ , in (2.18) and the clock dynamics in (2.5). Here we exploit the fact that time interval between two consecutive Poincaré sampling events is equal to the period of the controller clock,  $T_c$ ,

$$T_c = \underbrace{\beta_{[1,e-1]} + \varphi_e(\alpha)}_{\mathcal{G}_{[1,e]}} + \underbrace{T_{\mathcal{A}}(\mathbf{p}_k)}_{\mathcal{A}_{[e,N]}} + \underbrace{\alpha_{k+1}}_{\mathcal{G}_N}. \quad (2.27)$$

We will individually consider each additive term of the right hand side in the above equation and derive them in terms of the Poincaré state,  $\nu$  and  $\alpha$ . The total duration of the cells completely swept during stance is a constant and given by

$$\beta_{[1,e-1]} := \sum_{j=1}^{e-1} [\beta_j]. \quad (2.28)$$

By direct computation we obtain the remaining stance time — time spent in the ground contact during the  $i^{\text{th}}$  cell — as

$$\varphi_e(\alpha) := \mathbf{R}_e \circ \mathbf{b}_{e-1}^e \circ \phi_e^*(\alpha). \quad (2.29)$$

where  $\mathbf{R}_e(s) := (1/\gamma_e)(\phi_{LO}^e - s)$ . Note that the remaining time function,  $\varphi_e(\alpha)$ , is a strictly monotonically decreasing function and has a well-defined inverse. The flight duration,  $T_{\mathcal{A}}$ , is a linear function of the lift-off speed. Since the lift-off and touchdown occur at the same height the lift-off speed is identical to that of the next touchdown given by the energy component of the return map,  $\mathbf{r}_1(\mathbf{p})$ , in (2.26). Hence, we can compute the flight time in terms of the current Poincaré states,

$$T_{\mathcal{A}}(\mathbf{p}_k) := 2\nu_{k+1} = 2\nu_k \rho(\alpha_k) \quad (2.30)$$

By definition  $\alpha_k$  is the time delay between the last touchdown (which happens in  $(k-1)^{\text{th}}$  clock cycle) and the start of the  $k^{\text{th}}$  clock cycle. Similarly,  $\alpha_{k+1}$  is the time difference between the touchdown event in the  $k^{\text{th}}$  cycle at the end of the aerial phase and the conclusion of the  $k^{\text{th}}$  clock cycle. Substitution of each expression back to the cycle period equality and reorganization of the terms lead to the phase component of the return map,

$$\mathbf{r}_2(\mathbf{p}) = T_c - \beta_{[1,e-1]} - \varphi_e(\alpha) - 2\nu\rho(\alpha).$$

□

## 2.4.2 Mode Sequence Domain Derivation

The  $\mathcal{GAG}(e, N)$  mode sequence family offers distinct analytical advantages. Most importantly, the special form of their return map,  $\mathbf{r}(\mathbf{p})$ , in (2.22) explicitly separates the speed,  $\nu$ , and delay,  $\alpha$ , states, that is, each multiplicative and additive component in the return map expressions is a function of only one of the Poincaré states. Moreover, the Poincaré states,  $\nu$  and  $\alpha$ , are associated with physical quantities. This structure allows us to analytically derive the domain of each member mode sequence,  $\mathcal{P}(\sigma)$ , as stated in Proposition 2.

**Proposition 2.** *Let  $\sigma \in \mathcal{GAG}(e, N)$ . Its domain,  $\mathcal{P}(\sigma)$ , is a connected open subset of the Poincaré space,  $\mathcal{P}$ , given by*

$$\mathcal{P}(\sigma) := \{ (\alpha, \nu) \in \mathcal{P} \mid \alpha \in \mathcal{D}(\sigma) \wedge \pi_2 \circ \mathbf{r}(\mathbf{p}) \in (0, \max\{\beta_N, \pi/\gamma_N\}) \}. \quad (2.31)$$

where  $\mathcal{D}(\sigma)$  is the “valid delay interval” defined in (2.32).

PROOF. Physical interpretation of the delay state,  $\alpha$ , for the  $\mathcal{GAG}(e, N)$  family is the key in this derivation. For an operating regime of the coupled system in (2.6), characterized by a mode sequence,  $\sigma \in \mathcal{GAG}(e, N)$ , the timing of the four events (clock start, lift-off, touchdown and clock end) along its continuous-time trajectory,  $[\mathbf{x}(t) \ \theta(t)]^T$ , from one return to the next gives rise to a set of inequalities that must be satisfied by the Poincaré states,  $(\nu, \alpha)$ , identifying the corresponding domain of the mode sequence,  $\mathcal{P}(\sigma)$ . We will divide the derivations into two parts: bounds on the delay state; and bounds on the speed state. The reader can refer to Figure 2.7 as

a visual aid for the explanations.

We compute the bounds on the delay state,  $\alpha$ , using the clock start and lift-off conditions, both of which are independent of the speed state,  $\nu$ , resulting in an inequality only in terms of the delay state,  $\alpha$ .

The clock cycle must start during stance mode,  $\mathbf{x}(\theta = 0) \in \mathcal{G}$ , and therefore, in the preferred mechanical coordinates,  $\mathcal{E}_N$ , the mechanical phase,  $\phi$ , sampled at the clock reset,  $\theta = 0$ , must be in  $[\phi_{TD}^N, \phi_{LO}^N)$ . Hence, the delay state,  $\alpha$ , which is related to the mechanical phase by (2.20), must satisfy,  $\alpha \in [0, \pi/\gamma_N)$ .

For  $\mathcal{GAG}(e, N)$  mode sequence family the (only) lift-off event occurs in the  $e^{\text{th}}$  cell,  $\theta(\mathbf{x} \in \partial\mathcal{X}_G^+) \in \Theta_e$ , hence, the remaining time function,  $\varphi_e(\alpha)$ , defined in (2.29), expressing the time spent in ground contact during the  $e^{\text{th}}$  cell prior to lift-off must be non-negative and smaller than the duration of the  $e^{\text{th}}$  cell duration,  $\beta_e$ . Noting that  $\varphi_e(\alpha)$  is a strictly monotonically decreasing function of the delay,  $\alpha$ , we obtain a second inequality in terms of the delay state,  $\alpha \in (\varphi_e^{-1}(\beta_e), \varphi_e^{-1}(0))$ .

Combining the two inequalities above we define the “valid delay interval,”  $\mathcal{D}(\sigma)$ , of the mode sequence  $\sigma$ , given by

$$\mathcal{D}(\sigma) := (\varphi_e^{-1}(\beta_e), \varphi_e^{-1}(0)) \cap [0, \pi/\gamma_N). \quad (2.32)$$

The second part of the derivation focuses on the speed state,  $\nu$ , where we utilize the touchdown and clock end conditions. Our derivation identifies bounds that need to be satisfied by the time spent in stance mode,  $\mathcal{G}$ , during the  $N^{\text{th}}$  cell which can be written in terms of the Poincaré states,  $\pi_2 \circ \mathbf{r}(\mathbf{p})$ , according to (2.22).

The (only) touchdown event in the  $\mathcal{GAG}(e, N)$  mode sequence must occur in the  $N^{\text{th}}$  cell,  $\theta(\mathbf{x} \in \partial\mathcal{X}_G^-) \in \Theta_N$ , therefore, the time spent in the  $N^{\text{th}}$  cell during ground contact must be non-zero and smaller than the duration of the  $N^{\text{th}}$  cell,  $\beta_N$ , leading to the first bound,  $\pi_2 \circ \mathbf{r}(\mathbf{p}) \in (0, \beta_N)$ . Furthermore, the clock end must occur in

the ground mode,  $\mathbf{x}(\theta = 2\pi) \in \mathcal{G}$ . Following the same process as in the clock start condition yields a second bound,  $\pi_2 \circ \mathbf{r}(\mathbf{p}) \in [0, \pi/\gamma_N]$ .

Taking the intersection of these two bounds results in

$$\pi_2 \circ \mathbf{r}(\mathbf{p}) \in (0, \min \{\beta_N, \pi/\gamma_N\}).$$

We observe that both the speed,  $\nu$ , and the delay,  $\alpha$ , states appear in the resulting inequality. Note that the special structure of the delay component of the return map in (2.22) allows us to reorganize this bound as a delay,  $\alpha$ , parametrized bound on the speed state,

$$V(\alpha, 0) > \nu > V(\alpha, \min \{\beta_N, \pi/\gamma_N\})$$

where

$$V(\alpha, s) := \frac{T_c - \beta_{[1, e-1]} - \varphi_e(\alpha) - s}{2\rho(\alpha)}$$

Notice that the valid speed values are parameterized by the clock period,  $T_c$ . This feature of the valid domain,  $\mathcal{P}(\sigma)$ , will prove to be useful in later sections.

The domain of the mode sequence,  $\mathcal{P}(\sigma)$ , given in (2.31), is the intersection of these two connected open sets and therefore, itself is open and connected.

□

### 2.4.3 Derivation of Fixed Points

The fixed points,  $\mathbf{p}^*$ , of the return map,  $\mathbf{r}(\mathbf{p}^*)$ , in (2.22) correspond to the cyclic behaviors of the clock driven 1-DOF hopper operating in the associated fundamental repeatable mode sequence,  $\sigma \in \mathcal{GAG}(e, N)$ . Careful derivation of physically valid

fixed points,  $\mathbf{p}^*$ , is a prerequisite for the stability analysis of the coupled system that will be presented in the later section. This section will explain the derivation of fixed points,  $\mathbf{p}^*$ , of the return map,  $\mathbf{r}(\mathbf{p})$ , and discuss conditions that guarantee their physical relevance.

The special structure of the return map,  $\mathbf{r}(\mathbf{p})$ , in (2.22) allows us to reorganize the fixed point criterion,  $\mathbf{r}(\mathbf{p}^*) = \mathbf{p}^*$ , into a set of equations,

$$\begin{aligned}\rho(\alpha^*) &= 1, \\ \nu^* &= [T_c - \beta_{[1, e-1]} - \varphi_e(\alpha^*) - \alpha^*] / 2,\end{aligned}\tag{2.33}$$

revealing that each unity crossing of the forcing function,  $\rho(\alpha)$ , determines a fixed point,  $\mathbf{p}^*$ .

Defined by the unity crossings of the forcing function,  $\rho(\alpha)$ , the delay values of the fixed points,  $\alpha^*$ , are independent of the clock period,  $T_c$ , and only parametrized by the mechanical loss,  $\{\mu, \zeta\}$ , and controller shape parameters,  $\mathbf{o}$ . On the other hand, the fixed point speed,  $\nu^*$ , is an affine function of the clock period,  $T_c$ .

Our derivations will take advantage of this *effective independence* between the fixed point speed,  $\nu^*$ , and the clock period,  $T_c$ , which allows us to alter the fixed point speed,  $\nu^*$ , without changing the fixed point delay,  $\alpha^*$ , by simply varying the clock period,  $T_c$ . In fact, in the following analysis it will prove to be very useful to treat the fixed point speed,  $\nu^*$ , and the fixed point delay,  $\alpha^*$ , as if they were two independent variables.

The return map,  $\mathbf{r}_\sigma(\mathbf{p})$ , of a mode sequence,  $\sigma \in \mathcal{M}$ , is physically relevant only over the domain of the mode sequence,  $\mathcal{P}(\sigma)$ . Hence, our stability analysis has to consider only those solutions of the fixed point criterion,  $\mathbf{r}_\sigma(\mathbf{p}^*) = \mathbf{p}^*$ , that fall within this subset of the Poincaré space. We will refer to these solutions as the “valid fixed points” of the return map and denote the set of them by

$$\mathcal{P}^* := \{\mathbf{p}^* \in \mathcal{P}(\sigma) \mid \mathbf{r}_\sigma(\mathbf{p}^*) = \mathbf{p}^*\}$$

For the fundamental repeatable mode sequences,  $\sigma \in \mathcal{GAG}(e, N)$ , one can check the validity of a fixed point,  $\mathbf{p}^*$ , based on the location of its delay state,  $\alpha^*$ , and the value of the clock period,  $T_c$ , as explained in Lemma 2.

**Lemma 2.** *Let  $\mathbf{r}(\mathbf{p})$  be the return map of a fundamental repeatable mode sequence,  $\sigma \in \mathcal{GAG}(e, N)$ . A fixed point,  $\mathbf{p}^*$ , is “valid,” if and only if, its delay state,  $\alpha^*$ , is in the “invariant delay interval,”  $\alpha^* \in \mathcal{D}^*(\sigma)$ ,*

$$\mathcal{D}^*(\sigma) := (0, \min\{\beta_N, \pi/\gamma_N\}) \cap \mathcal{D}(\sigma), \quad (2.34)$$

and the clock period is in the “admissible period interval,”

$$T_c \in \mathcal{T}(\alpha^*) := (T_c^0(\alpha^*), \infty),$$

where

$$T_c^0(\alpha^*) := \beta_{[1, e-1]} + \varphi_e(\alpha^*) + \alpha^* \quad (2.35)$$

is the “smallest admissible period.”

PROOF. By definition, a fixed point,  $\mathbf{p}^*$ , is valid, if and only if, it resides within the domain of the mode sequence,  $\mathcal{P}(\sigma)$ . According to (2.31) states of the fixed point must satisfy three conditions: 1)  $\alpha^* \in \mathcal{D}(\sigma)$ ; 2)  $\pi_2 \circ \mathbf{r}(\mathbf{p}^*) \in (0, \min\{\beta_N, \pi/\gamma_N\})$ ; and 3)  $\nu^* > 0$ . The discussion below will refer to these conditions as the “validity conditions.” Our derivations will be presented in two parts: derivation of the invariant delay interval,  $\mathcal{D}^*(\sigma)$ ; and derivation of the admissible clock interval,  $\mathcal{T}(\alpha^*)$ .

We will start with the derivation of the invariant delay interval,  $\mathcal{D}^*(\sigma)$ . Using the explicit form of the return map,  $\mathbf{r}(\mathbf{p})$ , in (2.22) we can rewrite the second validity condition as a set of inequalities,

$$\frac{T_c - \beta_{[1,e-1]} - \varphi_e(\alpha^*)}{2} > \nu^* > \frac{T_c - \beta_{[1,e-1]} - \varphi_e(\alpha^*) - \min\{\beta_N, \pi/\gamma_N\}}{2},$$

which can be further simplified by substituting the fixed point speed,  $\nu^*$ , expression from (2.33) leading into a set of inequalities only in terms of the fixed point delay,  $\alpha^*$ ,

$$0 < \alpha^* < \min\{\beta_N, \pi/\gamma_N\}.$$

The invariant delay interval,  $\mathcal{D}^*(\sigma)$ , given in (2.34), is the intersection between the intervals defined by the first validity condition and this equivalent form of the second.

Next, we will consider the third validity condition,  $\nu^* > 0$ , which can be rewritten, by substituting the fixed point speed expression from (2.33), as an inequality of a given fixed point delay,  $\alpha^*$ , and the clock period,  $T_c$ ,

$$(T_c - \beta_{[1,e-1]} - \varphi_e(\alpha^*) - \alpha^*)/2 > 0.$$

Since the fixed point delay,  $\alpha^*$ , is independent of the clock period,  $T_c$ , this inequality effectively defines an interval on the clock period space,  $\mathcal{T}(\alpha^*)$ , given in (2), that is parametrized by the given fixed point delay,  $\alpha^*$ . It directly follows that the third validity condition,  $\nu^* > 0$ , is satisfied by all clock periods,  $T_c$ , in this interval,  $\mathcal{T}(\alpha^*)$  which we refer as the admissible period interval. The smallest admissible period,  $T_c^0(\alpha^*)$ , which is the lower bound of the admissible period interval, can be directly computed from above inequality.

□

## 2.4.4 Local Stability Analysis

This section concerns the (local) stability analysis of the return map,  $\mathbf{r}(\mathbf{p})$ , in (2.22) which captures the behavior of the clock driven 1-DOF hopper operating in fundamental repeatable mode sequences,  $\sigma \in \mathcal{GAG}(e, N)$ . The analysis will aim to identify (sufficient) conditions that give rise to asymptotically stable fixed points,  $\mathbf{p}^*$ .

It follows from the definition of fixed points in (2.33) and from Lemma 2 that the existence of valid fixed points is a function of the shape configuration,  $\mathbf{o}$ . For the immediate stability discussions we will assume that there exists a set of valid fixed points,  $\mathcal{P}^* \neq \emptyset$ , and postpone the investigation of the conditions that lead to existence of valid fixed points to a later section. Theorem 1 states the main stability result.

**Theorem 1.** *A clock controlled 1-DOF hopper operating in a fundamental repeatable mode sequence,  $\sigma \in \mathcal{GAG}(e, N)$ , has a locally asymptotically stable limit cycle (period-one hopping gait) if: 1) the forcing function has a unity crossing,  $\rho(\alpha^*) = 1$ , in the invariant delay interval of the mode sequence,  $\alpha^* \in \mathcal{D}^*(\sigma)$ , defined in (2.34), such that  $\rho'(\alpha^*) > 0$ ; 2) the mechanical hopper is lossy,  $\mu > 0$ ,  $\zeta \in (0, 1)$ ; and 3) the clock period is in the interval of stable periods,  $T_c \in \mathcal{T}_s(\alpha^*)$ , which is defined in (2.49).*

PROOF. The theorem follows directly from the results in Proposition 3, Proposition 4, Proposition 5 and Lemma 8 which will be stated and proved below. □

To characterize the (local) stability properties of a given fixed point,  $\mathbf{p}^*$ , we derive the Jacobian of the return map,  $\mathbf{r}(\mathbf{p})$ , evaluated at this fixed point,

$$\mathbf{J}(\mathbf{p}^*) := D\mathbf{r}(\mathbf{p})|_{\mathbf{p}=\mathbf{p}^*} = \begin{bmatrix} 1 & \nu^* \rho'(\alpha^*) \\ -2 & -\varphi_e'(\alpha^*) - 2\nu^* \rho'(\alpha^*) \end{bmatrix}, \quad (2.36)$$

and compute its two eigenvalues,

$$\lambda_{\pm} := 1/2(\text{tr} \pm \sqrt{\Delta}), \quad (2.37)$$

that we choose to write in terms of the trace,

$$\text{tr} := 1 - \varphi_e'(\alpha^*) - 2\nu^* \rho'(\alpha^*), \quad (2.38)$$

and the discriminant,

$$\Delta := [1 - \varphi_e'(\alpha^*) - 2\nu^* \rho'(\alpha^*)]^2 + 4\varphi_e'(\alpha^*), \quad (2.39)$$

of the Jacobian,  $\mathbf{J}(\mathbf{p}^*)$ .

In essence, our stability analysis will take the form of a root-locus study of the Jacobian,  $\mathbf{J}(\mathbf{p}^*)$ , aiming to determine conditions on the coupled system parameters, the shape configuration,  $\mathbf{o}$ , the clock period,  $T_c$ , and dissipation parameters,  $\{\mu, \zeta\}$ , that places the eigenvalues,  $\lambda_{\pm}$ , inside the unit circle rendering the fixed point,  $\mathbf{p}^*$ , locally asymptotically stable. However, direct computation of the eigenvalues,  $\lambda_{\pm}$ , as a function of these parameters,  $\{T_c, \mathbf{o}, \mu, \zeta\}$ , yields complex expressions, and therefore, it is not a suitable means to extract compact stability conditions. Instead, we will divide the analysis into sub-problems that are easier to investigate and combine their individual results to form the global result stated in Theorem 1.

We observe that the eigenvalue expressions in (2.37)-(2.39) contain three terms: the first derivative of the forcing function evaluated at the fixed point delay,  $\rho'(\alpha^*)$ ; the first derivative of the remaining time function evaluated at the fixed point delay,  $\varphi_e'(\alpha^*)$ ; and the fixed point speed,  $\nu^*$ . It is important to note that in the eigenvalue

expressions the fixed point speed,  $\nu^*$ , is the only term that is parameterized by the clock period,  $T_c$ . The remaining terms,  $\rho'(\alpha^*)$  and  $\varphi_e'(\alpha^*)$ , (like the fixed point delay,  $\alpha^*$ , itself) depend only on the dissipation,  $\{\mu, \zeta\}$ , and shape configuration,  $\mathbf{o}$ . For the following discussions we will treat the trace,  $\text{tr}(T_c)$ , and the discriminant,  $\Delta(T_c)$ , as functions of clock period,  $T_c$ , that are parameterized by system parameters,  $\{\mu, \zeta, \mathbf{o}\}$ , and the fixed point delay<sup>5</sup>,  $\alpha^*$ . For notational simplicity in the following discussion the period dependency of the trace,  $\text{tr}$ , and the discriminant,  $\Delta$ , will not be explicitly stated.

According to the sign of  $\rho'(\alpha^*)$  — a scalar valued function of the dissipation parameters,  $\{\mu, \zeta\}$ , and the shape configuration,  $\mathbf{o}$  — we partition the set of fixed points into three classes,  $\mathcal{P}^* = \mathcal{P}_-^* \cup \mathcal{P}_0^* \cup \mathcal{P}_+^*$ ,

$$\begin{aligned}\mathcal{P}_-^* &:= \{\mathbf{p}^* \in \mathcal{P}^* \mid \rho'(\alpha^*) < 0\} \\ \mathcal{P}_0^* &:= \{\mathbf{p}^* \in \mathcal{P}^* \mid \rho'(\alpha^*) = 0\} \\ \mathcal{P}_+^* &:= \{\mathbf{p}^* \in \mathcal{P}^* \mid \rho'(\alpha^*) > 0\}.\end{aligned}\tag{2.40}$$

Since both the fixed point delay,  $\alpha^*$ , as well as the related terms,  $\rho'(\alpha^*)$  and  $\varphi_e'(\alpha^*)$ , are independent of the clock period,  $T_c$ , varying the clock period,  $T_c$ , moves the eigenvalues,  $\lambda_{\pm}$ , of the Jacobian,  $\mathbf{J}(\mathbf{p}^*)$ , along one dimensional (easily computable) curves in the complex plane while keeping which class the fixed point,  $\mathbf{p}^*$ , belongs unchanged. This property allows us to separately conduct root-locus analysis for each class as a function of the clock period,  $T_c$ , and identify their stability properties.

To aid our explanations we define a discriminant bound,  $\overline{\Delta} := \Delta(T_c^0(\alpha^*))$ ,

$$\begin{aligned}\overline{\Delta} &= (1 - \varphi_e'(\alpha^*))^2 + 4\varphi_e'(\alpha^*) \\ &= (1 + \varphi_e'(\alpha^*))^2,\end{aligned}\tag{2.41}$$

---

<sup>5</sup>In fact, the fixed point delay,  $\alpha^*$ , is a function of the system parameters,  $\{\mu, \zeta, \mathbf{o}\}$ , according to (2.33). However, in our stability discussions it will prove to be useful to think of the fixed point delay,  $\alpha^*$ , as a separate parameter.

and a trace bound,  $\overline{\text{tr}} := \text{tr}(T_c^0(\alpha^*))$ ,

$$\overline{\text{tr}} = 1 - \varphi_e'(\alpha^*), \quad (2.42)$$

where  $T_c^0(\alpha^*)$  is the smallest admissible clock period defined in (2.35). Notice that the discriminant bound is strictly positive definite,  $\overline{\Delta} > 0$ , and the trace bound is greater than unity,  $\overline{\text{tr}} > 1$ , since the remaining time function is strictly decreasing,  $\varphi_e'(\alpha) < 0$ , by definition. In the following subsections we will look into each fixed point class separately and determine its stability properties.

### Stability Properties of $\mathcal{P}_-$

To assess stability we will compute the locus of the eigenvalues for all fixed points,  $\mathbf{p}^*$ , in  $\mathcal{P}_-$  and check where they are located relative the unit circle. To this end in Lemma 3 we will first identify the bounds on the discriminant,  $\Delta$ , and the trace,  $\text{tr}$ , of the Jacobian,  $\mathbf{J}(\mathbf{p}^*)$ , limited to the fixed points,  $\mathbf{p}^*$ , in  $\mathcal{P}_-$ .

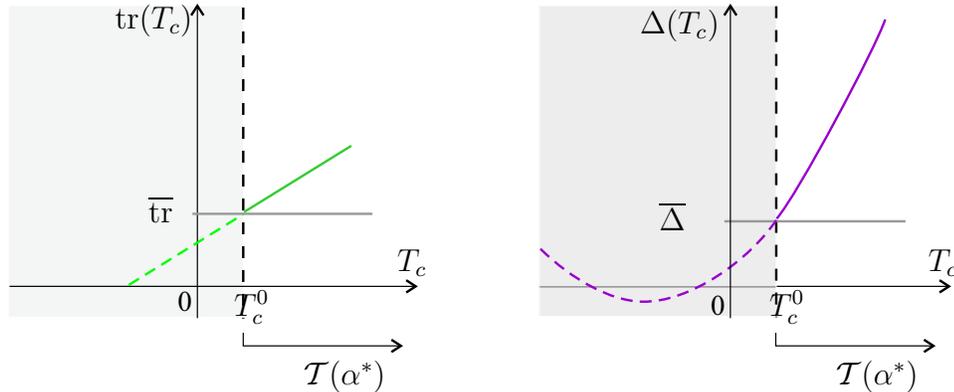


Figure 2.8: The trace,  $\text{tr}$ , (left) and the discriminant,  $\Delta$ , (right) functions and their respective bounds plotted against the clock period,  $T_c$ , for a given fixed point,  $\mathbf{p}^*$ , in  $\mathcal{P}_-$ . The physically uninteresting interval of the clock period axis is shaded.

**Lemma 3.** *Let  $\mathbf{p}^*$ , be a fixed point in  $\mathcal{P}_-$ . For the associated Jacobian,  $\mathbf{J}(\mathbf{p}^*)$ , both the discriminant,  $\Delta$ , and the trace,  $tr$ , are strictly monotonically increasing functions of the clock period,  $T_c$ , and strictly greater than their respective bounds over the interval of admissible periods,  $T_c \in \mathcal{T}(\alpha^*)$ ,*

$$\forall \mathbf{p}^* \in \mathcal{P}_- \quad \forall T_c \in \mathcal{T} \quad (\Delta > \bar{\Delta} > 0 \wedge tr > \bar{tr} > 1) \quad \wedge \quad \left( \frac{d\Delta}{dT_c} > 0 \wedge \frac{dtr}{dT_c} > 0 \right).$$

PROOF. The reader can refer to Figure 2.8 as a visual aid for this derivation. The difference between the discriminant,  $\Delta$ , and its bound,  $\bar{\Delta}$ ,

$$\Delta - \bar{\Delta} = \nu^* \left[ \nu^* - \frac{1 - \varphi_e'(\alpha^*)}{2\rho'(\alpha^*)} \right] [2\rho'(\alpha^*)]^2,$$

is a convex quadratic function of the fixed point speed,  $\nu^*$ , with two real roots,

$$\nu_1^* = 0, \quad \nu_2^* = \frac{1 - \varphi_e'(\alpha^*)}{2\rho'(\alpha^*)}.$$

whose non-zero root is strictly negative definite,  $\nu_2^* < 0$ , since by hypothesis  $\rho'(\alpha^*) < 0$  (for all  $\mathbf{p}^* \in \mathcal{P}_-$ ) and the remaining time function is a strictly decreasing function,  $\varphi_e'(\alpha) < 0$ , by definition. The difference between the discriminant,  $\Delta$ , and its bound,  $\bar{\Delta}$ , is negative between the two roots,  $(\nu_1^*, \nu_2^*)$ , and positive elsewhere, and therefore, the discriminant,  $\Delta$  is larger than its bound,  $\bar{\Delta}$ , for all positive definite fixed point speed values,  $\nu^* > 0$ . It follows directly from the relationship between the fixed point speed,  $\nu^*$ , and the clock period,  $T_c$ , in (2.33) that the discriminant is larger than its bound,  $\Delta > \bar{\Delta}$ , for all  $T_c > T_c^0$ .

Next, we look at the difference between the trace,  $tr$ , and its bound,  $\bar{tr}$ ,

$$tr - \bar{tr} = -2\nu^* \rho'(\alpha^*),$$

which is a linear function of the fixed point speed,  $\nu^*$ , that is strictly increasing for all  $\mathbf{p}^* \in \mathcal{P}_-$  since  $\rho'(\alpha^*) < 0$ . Thus, the difference is strictly positive for all strictly positive (physically relevant) fixed point speed,  $\nu^* > 0$ . Again, using the relationship in (2.33) we conclude that the trace is strictly larger than its bound,  $\text{tr} > \overline{\text{tr}}$ , for all  $T_c > T_c^0(\alpha^*)$ .

Both trace,  $\text{tr}$ , and the discriminant,  $\Delta$ , are monotonically increasing functions of the fixed point speed,  $\nu^*$ , which is an affine function of the clock period according to (2.33). Hence, both functions are monotonically increasing functions of the clock period,  $T_c$ . See Figure 2.8.

□

It directly follows from Lemma 3 and the eigenvalue expression in (2.37)-(2.39) that for fixed points,  $\mathbf{p}^*$ , in  $\mathcal{P}_-$ , the locus of the two eigenvalues,  $\lambda_{\pm}$ , depicted in Figure 2.9, always contain one eigenvalue outside the unit circle and the corresponding fixed points,  $\mathbf{p}^*$ , are therefore unstable. A detailed explanation of this statement can be found in Proposition 3.

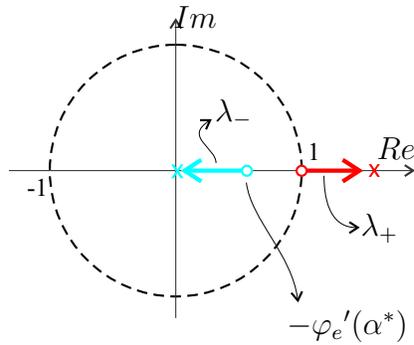


Figure 2.9: Eigenvalues,  $\lambda_{\pm}$ , of the Jacobian evaluated at a fixed point,  $\mathbf{p}^*$  in  $\mathcal{P}_-$  as a function of the clock period,  $T_c$ , over the admissible clock period values,  $T_c \in (T_c^0(\alpha^*), \infty)$ . Circles indicate the location of the eigenvalues at  $T_c = T_c^0(\alpha^*)$ .

**Proposition 3.**  $\mathcal{P}_-^*$  is a collection of unstable fixed points.

PROOF. Consider a member of the fixed point class of interest,  $\mathbf{p}^* \in \mathcal{P}_-^*$ . According to Lemma 3, for all associated admissible clock periods,  $T_c > T_c^0(\alpha^*)$ , the Jacobian of the return map evaluated at this point,  $\mathbf{J}(\mathbf{p}^*)$ , has a positive definite discriminant,  $\Delta > \bar{\Delta} > 0$ , and therefore, the eigenvalues of the Jacobian are real,  $\lambda_{\pm} \in \mathbb{R}$ . By direct computation we can show that a lower bound places  $\lambda_+$  outside the unit circle,  $\lambda_+ = \frac{1}{2}(\text{tr} + \sqrt{\Delta}) > \frac{1}{2}(\bar{\text{tr}} + \sqrt{\bar{\Delta}}) = 1$ . Hence, we conclude that all fixed points,  $\mathbf{p}^*$ , in  $\mathcal{P}_-^*$  are unstable. □

### Stability Properties of $\mathcal{P}_0^*$

According to (2.40) the fixed points,  $\mathbf{p}^* \in \mathcal{P}_0^*$ , are defined by unity crossings of the forcing function,  $\rho(\alpha)$ , at which its first derivative vanishes,  $\rho'(\alpha^*) = 0$ . Unfortunately, for this class, both eigenvalues are located on the unit circle,  $\lambda_{\pm} = 1$ , which does not allow us to assess the local stability properties of the nonlinear discrete return map,  $\mathbf{r}(\mathbf{p})$ .

For the fixed points,  $\mathbf{p}^*$ , in  $\mathcal{P}_0^*$ , there are two possible scenarios: 1) the forcing function,  $\rho(\alpha)$ , has an inflection point at the fixed point delay,  $\alpha^*$ ; or 2) the forcing function,  $\rho(\alpha)$ , is tangent to unity at a local extremum (maximum or minimum) at the fixed point delay,  $\alpha^*$ . Below we will consider both scenarios and provide incomplete arguments suggesting that fixed points of this class are unstable as stated in Conjecture 1.

**Conjecture 1.**  $\mathcal{P}_0^*$  is a collection of unstable fixed points.

Our numerical studies suggests that the forcing function,  $\rho(\alpha)$ , does not have any inflection points. Although we cannot present an analytic proof at this point. We will conjecture this property of the forcing function in Conjecture 2 and ignore this possibility.

**Conjecture 2.** *For all shape configurations,  $\mathbf{o} \in \mathcal{O}$ , the forcing function,  $\rho(\alpha)$ , has no inflection point,*

$$\forall \mathbf{o} \in \mathcal{O} \quad \nexists \alpha \in \mathcal{D} \quad \text{s.t.} \quad (\rho'(\alpha) = 0 \wedge \rho''(\alpha) = 0)$$

On the other hand, the bounded forcing function,  $\rho(\alpha)$ , does have local extrema which can be tangent to unity for properly adjusted system parameters,  $\{\mu, \zeta, \mathbf{o}\}$ . Lemma 4 guarantees that those fixed points,  $\mathbf{p}^*$ , defined by the tangents of the forcing function,  $\rho(\alpha)$ , with unity at such extrema are not stable.

**Lemma 4.** *Let the forcing function  $\rho(\alpha)$  have a local extremum at  $\alpha^*$  where it is tangent to unity,  $\rho(\alpha^*) = 1$ . The corresponding fixed point,  $\mathbf{p}^*$ , is not stable.*

PROOF. Recall that the forcing function,  $\rho(\alpha)$ , determines the change in the speed state,  $\nu$ , according to (2.22). Suppose the forcing function,  $\rho(\alpha)$ , has a local maximum at  $\alpha^*$  where it is tangent to unity. By definition the forcing function is strictly less than unity in an open neighborhood of this delay value,

$$\exists r > 0 \quad \text{s.t.} \quad \forall \alpha \in B_r(\alpha^*) \quad \rho(\alpha) < 1$$

Consider an arbitrarily small disturbance to the fixed point,  $\mathbf{p}^*$ , such that the the fixed point speed is shifted in the negative direction,  $\mathbf{p}_0 = \mathbf{p}^* - [\epsilon, 0]^T$ . It directly follows from the return map definition in (2.22) that the delay state at the next

return is displaced away from the fixed point delay,  $\alpha_1 \neq \alpha^*$ . Since the forcing function is smaller than the unity in the neighborhood the Poincaré sequence,  $\{\mathbf{p}_k\}_{k=0}^{\infty}$ , will diverge away from the fixed point,  $\mathbf{p}^*$ . Hence, there does not exist any open neighborhood about the fixed point where all Poincaré sequences converge to the fixed point, that is, the fixed point,  $\mathbf{p}^*$ , is not stable.

A similar argument can be made for the case where the local extremum at  $\alpha^*$  is a local minimum. □

### Stability Properties of $\mathcal{P}_+^*$

Next, we will study the stability properties of the fixed points,  $\mathbf{p}^*$ , in  $\mathcal{P}_+^*$ . Limiting ourselves to this class of fixed points we will first derive the properties of the trace,  $\text{tr}$ , and the discriminant,  $\Delta$  in Lemma 5 and Lemma 6, respectively. The reader can refer to Figure 2.10 as a visual aid which illustrates typical forms of the trace,  $\text{tr}$ , and the discriminant,  $\Delta$ , for  $\mathbf{p}^* \in \mathcal{P}_+^*$ .

**Lemma 5.** *Let  $\mathbf{p}^*$  be a fixed point in  $\mathcal{P}_+^*$ . For the associated Jacobian,  $\mathbf{J}(\mathbf{p}^*)$ , the trace,  $\text{tr}$ , is a strictly monotonically decreasing function of the clock period,  $T_c$ , and is strictly less than its bound,  $\overline{\text{tr}}$ , over the admissible period interval,  $\mathcal{T}$ ,*

$$\forall T_c \in \mathcal{T} \quad \text{tr} < \overline{\text{tr}}$$

PROOF. Using the definition of the trace,  $\text{tr}$ , in (2.38) and the relationship between the fixed point speed,  $\nu^*$ , and the clock period,  $T_c$ , given by (2.33) we compute the derivative of the trace with respect to the clock period,

$$\frac{d\text{tr}}{dT_c} := -\rho'(\alpha^*),$$

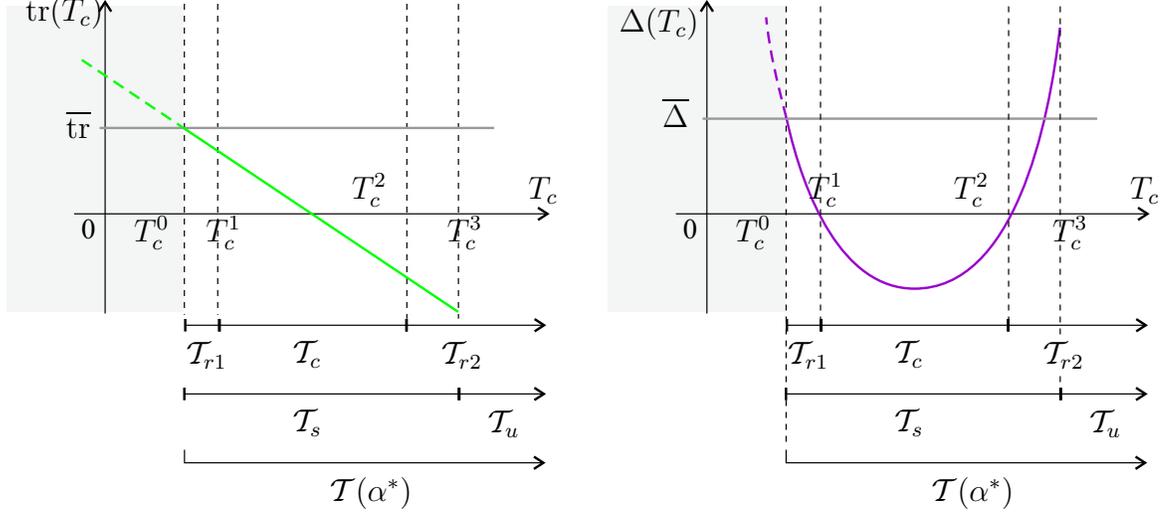


Figure 2.10: The trace,  $\text{tr}$ , (left) and the discriminant,  $\Delta$ , (right) functions and their respective bounds plotted against the clock period,  $T_c$ , for a given fixed point,  $\mathbf{p}^*$ , in  $\mathcal{P}_+^*$ . The physically uninteresting interval of the clock period axis is shaded.

which is negative definite for all  $\mathbf{p}^* \in \mathcal{P}_+^*$  by definition in (2.40). Hence, the trace,  $\text{tr}$ , is a strictly monotonically decreasing function of clock period,  $T_c$ .

The difference between the trace and its bound,  $\text{tr} - \bar{\text{tr}} = -2\nu^* \rho'(\alpha^*)$ , is linear in fixed point speed,  $\nu^*$ , and parametrized by the the first derivative of the forcing function evaluated at the fixed point delay,  $\rho'(\alpha^*)$ , which is positive definite for all  $\mathbf{p}^* \in \mathcal{P}^+$  according to (2.40). By definition the fixed point speed is positive definite,  $\nu^* > 0$ , over the admissible period interval,  $\mathcal{T}$ . Hence, we conclude that the difference is negative definite, and therefore, the trace,  $\text{tr}$ , is less than its bound,  $\bar{\text{tr}}$ , for all admissible clock periods,  $T_c \in \mathcal{T}$ , for all fixed points,  $\mathbf{p}^*$ , in  $\mathcal{P}_+^*$ .

□

**Lemma 6.** *Let  $\mathbf{p}^*$  be a fixed point in  $\mathcal{P}_+^*$ . For the associated Jacobian,  $\mathbf{J}(\mathbf{p}^*)$ ,*

the discriminant,  $\Delta$ , is a convex quadratic function of the clock period,  $T_c$ , whose two roots,  $\{T_c^1(\alpha^*), T_c^2(\alpha^*)\}$ , partition the admissible period interval, defined in (2), into three cells,  $\mathcal{T}(\alpha^*) := \mathcal{T}_{r1}(\alpha^*) \cup \mathcal{T}_c(\alpha^*) \cup \mathcal{T}_{r2}(\alpha^*)$ ,

$$\begin{aligned} \mathcal{T}_{r1}(\alpha^*) &:= (T_c^0(\alpha^*), T_c^1(\alpha^*)], \\ \mathcal{T}_c(\alpha^*) &:= (T_c^1(\alpha^*), T_c^2(\alpha^*)), \\ \mathcal{T}_{r2}(\alpha^*) &:= [T_c^2(\alpha^*), \infty). \end{aligned} \tag{2.43}$$

The discriminant is negative definite,  $\Delta < 0$ , for all  $T_c \in \mathcal{T}_c(\alpha^*)$ , and non-negative,  $\Delta \geq 0$ , otherwise,  $T_c \in \mathcal{T}_{r1}(\alpha^*) \cup \mathcal{T}_{r2}(\alpha^*)$ .

PROOF. The discriminant,  $\Delta$ , in (2.39) is a convex quadratic function of the fixed point speed,  $\nu^*$ . Its roots,

$$\nu_{\pm}^* = \frac{(1 - \varphi_e'(\alpha^*)) \pm 2\sqrt{-\varphi_e'(\alpha^*)}}{2\rho'(\alpha^*)} = \frac{(1 \pm \sqrt{-\varphi_e'(\alpha^*)})^2}{2\rho'(\alpha^*)}.$$

are real and positive definite,  $\nu_{\pm}^* \in \mathbb{R}^+$ , because by definition the remaining time function is strictly decreasing,  $\varphi_e'(\alpha) < 0$ , and by hypothesis the first derivative of the forcing function is positive definite,  $\rho'(\alpha^*) > 0$ , for all  $\mathbf{p}^* \in \mathcal{P}_+^*$ .

The fixed point speed,  $\nu^*$ , is a monotonically increasing affine function of the clock period,  $T_c$ , given by (2.33), therefore, the discriminant,  $\Delta$ , is a convex function of the clock period,  $T_c$ , whose two roots,

$$T_c^1(\alpha^*) := \nu^{*-1}(\nu_-^*) \quad \text{and} \quad T_c^2(\alpha^*) := \nu^{*-1}(\nu_+^*),$$

are in the admissible period interval,  $\{T_c^1(\alpha^*), T_c^2(\alpha^*)\} \in \mathcal{T}(\alpha^*)$ , since  $\nu_{\pm}^* \in \mathbb{R}^+$ . The roots of the discriminant,  $\{T_c^1(\alpha^*), T_c^2(\alpha^*)\}$ , and the smallest admissible period,  $T_c^0(\alpha^*)$ , define a partition of the admissible period interval given in (2.43).

It directly follows that the discriminant is negative definite,  $\Delta < 0$ , if  $T_c \in \mathcal{T}_c(\alpha^*)$ , and the discriminant is non-negative,  $\Delta \geq 0$ , if  $T_c \in \mathcal{T}_{r1}(\alpha^*) \cup \mathcal{T}_{r2}(\alpha^*)$ .

□

The admissible period interval,  $\mathcal{T}(\alpha^*)$ , as well as its cells,  $\mathcal{T}_{r1}(\alpha^*)$ ,  $\mathcal{T}_c(\alpha^*)$  and  $\mathcal{T}_{r2}(\alpha^*)$ , are functions of the fixed point delay,  $\alpha^*$ . However, for the sake of notational simplicity we will not state this relationship explicitly in the following discussions.

We will derive the locus of the eigenvalues,  $\lambda_{\pm}$ , as a function of the clock period,  $T_c$ , restricted to the admissible period interval,  $\mathcal{T}$ . We will separately compute the eigenvalues over each cell of the admissible period interval defined in (2.43) and use the continuity of the eigenvalues to combine these results into an overall eigenvalue locus which is illustrated in Figure 2.11 to aid the reader.

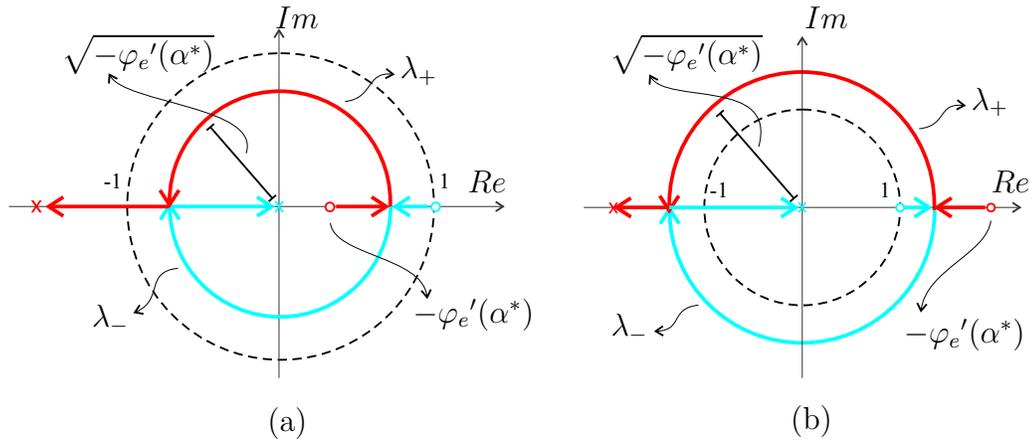


Figure 2.11: The eigenvalues of the Jacobian,  $\mathbf{J}(\mathbf{p}^*)$ , evaluated at a fixed point,  $\mathbf{p}^*$ , in  $\mathcal{P}_+^*$ . The locus is parametrized by the magnitude of  $\varphi_e'(\alpha^*)$  which determines whether the eigenvalues are (a) inside or (b) outside the unit circle, determining, in turn the stability of the fixed point,  $\mathbf{p}^*$ .

We will start off with the first cell of the admissible period interval,  $\mathcal{T}_{r1}$ , where the

eigenvalues of the Jacobian are real,  $\lambda_{\pm} \in \mathbb{R}$ . By definition at the smallest admissible period,  $T_c = T_c^0$ , both the trace,  $\text{tr}$ , and the discriminant,  $\Delta$ , are equal to their respective bounds, given in (2.42) and (2.41), allowing us to directly compute the eigenvalue expressions at  $T_c = T_c^0$ ,

$$\lambda_+(T_c^0) = \frac{1}{2} \left( \overline{\text{tr}} + \sqrt{\overline{\Delta}} \right) = 1, \quad \lambda_-(T_c^0) = \frac{1}{2} \left( \overline{\text{tr}} - \sqrt{\overline{\Delta}} \right) = -\varphi_e'(\alpha^*).$$

As the clock period,  $T_c$ , increases the two eigenvalues,  $\lambda_{\pm}$ , converge toward each other. At the upper boundary of the first cell,  $T_c = T_c^1$ , the discriminant vanishes,  $\Delta(T_c^1) = 0$ , and the eigenvalues become identical,  $\lambda_{\pm}(T_c^1) = \frac{1}{2}\text{tr}(T_c^1)$ .

In the second cell,  $\mathcal{T}_c$ , the discriminant is negative,  $\Delta < 0$ , and therefore, the eigenvalues are complex,  $\lambda_{\pm} \in \mathbb{C}$ , and conjugates of each other,  $\lambda_+ = \overline{\lambda_-}$ . The magnitude of the eigenvalues, which specifies their distance to the origin,

$$|\lambda_{\pm}| = \frac{1}{2} \sqrt{\text{tr}^2 - \Delta} = \sqrt{-\varphi_e'(\alpha^*)}, \quad (2.44)$$

is independent of clock period,  $T_c$ . Hence, for  $T_c \in \mathcal{T}_c$  the eigenvalues,  $\lambda_{\pm}$ , are located on a circle about the origin whose radius is  $\sqrt{-\varphi_e'(\alpha^*)}$ . It follows the continuity of the eigenvalues,  $\lambda_{\pm}$ , that at the lower bound of the current cell,  $T_c = T_c^1$ , the eigenvalues are located at the intersection between the positive real axis and this circle,

$$\lambda_{\pm}(T_c^1) = \sqrt{-\varphi_e'(\alpha^*)}.$$

The trace,  $\text{tr}$ , constitutes the real part of the complex conjugate eigenvalues and monotonically decreases with increasing period according to Lemma 5. Hence, the two eigenvalues move toward the left hand plane as  $\lambda_+$  and  $\lambda_-$  follow the upper and lower half circular paths, respectively. At  $T_c = T_c^2$  discriminant vanishes,  $\Delta = 0$ , placing the eigenvalues back on the real axis. We can use the continuity of the eigenvalues again to conclude that at the upper bound of the current cell,  $T_c = T_c^2$ ,

the eigenvalues,  $\lambda_{\pm}$ , are located at the intersection between the circle and the negative real axis,

$$\lambda_{\pm}(T_c^2) = -\sqrt{-\varphi_e'(\alpha^*)}.$$

According to Lemma 6 in the third and the last cell,  $\mathcal{T}_{r2}$ , the discriminant is non-negative,  $\Delta \geq 0$ , and the eigenvalues are real,  $\lambda_{\pm} \in \mathbb{R}$ . Recalling that the trace,  $\text{tr}$ , is monotonically decreasing for all  $\mathbf{p}^* \in \mathcal{P}_+^*$  according to Lemma 5 one can show that as the period,  $T_c$ , increases,  $\lambda_-$  converges to  $-\infty$ , whereas,  $\lambda_+$  moves along the real axis toward the origin.

We notice that the first derivative of the remaining time function evaluated at the fixed point delay,  $\varphi_e'(\alpha^*)$ , determines the stability properties of the fixed point,  $\mathbf{p}^*$ . For  $\varphi_e'(\alpha^*) \in (-\infty, -1)$  there is at least one eigenvalue that is outside the unit circle for all admissible clock periods,  $T_c \in \mathcal{T}$ , and therefore, the fixed point,  $\mathbf{p}^*$ , is unstable. On the other hand, if  $\varphi_e'(\alpha^*) \in (-1, 0)$ , there exists an interval of clock periods<sup>6</sup> where the eigenvalues lie inside the unit circle, and the fixed point,  $\mathbf{p}^*$ , is asymptotically stable. Figure 2.11 depicts the two possible scenarios for the eigenvalue locus.

We will establish a relationship between the mechanical loss parameters,  $\{\mu, \zeta\}$ , and the stability properties of the fixed point,  $\mathbf{p}^*$ . To this end Lemma 7 will derive a key relationship between the first derivative of the remaining time function,  $\varphi_e'(\alpha)$ , and the forcing function,  $\rho(\alpha)$ , leading to the results in Corollary 1 and Corollary 2.

**Lemma 7.** *The first derivative of the remaining time function,  $\varphi_e(\alpha)$ , with respect to the delay,  $\alpha$ , is related to the action terms,  $l_i(\alpha)$ , by*

$$\frac{d\varphi_e(\alpha)}{d\alpha} = - \left( \prod_{j=1}^e l_j(\alpha) \right)^{-2} \quad (2.45)$$

---

<sup>6</sup>Proposition 4 will characterize this interval of clock periods that give rise to asymptotically stable operating regimes.

PROOF. We will use the chain rule to compute the derivative of the remaining stance time,  $\varphi_e(\alpha)$ , whose definition in (2.29) can be expanded as

$$\varphi_e(\alpha) = R_e \circ M_e^{-1} \circ \left[ \begin{array}{c} e \\ \odot \quad (M_j \circ b_{j-1}^j) \\ j = 1 \end{array} \right] \circ P(\alpha),$$

where the derivatives of the translation terms are constant,  $\frac{d}{ds}(R_e(s)) = -1/\gamma_e$  and  $\frac{d}{ds}(M_i(s)) = \frac{d}{ds}[M_i^{-1}(s)] = 1$  for all  $i \in \{1, \dots, N\}$ . By direct computation we obtain the derivative of the relation map phase component in terms of the relation map energy component,  $a_i^j(\phi)$ ,

$$\frac{d}{d\phi}(b_i^j(\phi)) = \frac{\gamma_j}{\gamma_i} (a_i^j(\phi))^{-2},$$

which leads to

$$\frac{d}{d\alpha} \left( \left[ \begin{array}{c} e \\ \odot \quad (M_j \circ b_{j-1}^j) \\ j = 1 \end{array} \right] \circ P(\alpha) \right) = \gamma_N \prod_{j=1}^e \left[ \frac{\gamma_j}{\gamma_{j-1}} \left( \frac{1}{a_{j-1}^j \circ \phi_j^*(\alpha)} \right)^2 \right]. \quad (2.46)$$

Recalling that  $\mathbf{h}_0^1 \equiv \mathbf{h}_N^1$  and substituting the action term expression,  $l_i(\alpha)$ , according to (2.25) into the above equation results in (2.45).

□

**Corollary 1.** *Evaluated at a fixed point delay,  $\alpha^*$ , the first derivative of the remaining time function,  $\varphi_e'(\alpha^*)$ , and the loss term of the forcing function,  $\epsilon(\alpha^*)$ , are related by*

$$\varphi_e'(\alpha^*) = -(\epsilon(\alpha^*))^2. \quad (2.47)$$

PROOF. According to (2.33) the forcing function evaluated at a fixed point delay,  $\alpha^*$ , must be unity,  $\rho(\alpha^*) = 1$ . Using the definition of the forcing function,  $\rho(\alpha)$ , in (2.26) this condition leads to an equation in terms of the loss,  $\epsilon(\alpha^*)$ , and the action terms,  $l_j(\alpha^*)$ ,

$$\epsilon(\alpha^*) \left[ \prod_{j=1}^e l_j(\alpha^*) \right] = 1.$$

Taking squares of both sides of above equation and substitution of (2.45) results in (2.47).

□

Recall that the first derivative of the remaining time function evaluated at the fixed point delay,  $\varphi_e'(\alpha^*)$ , determines if the locus of the eigenvalues ever enter inside the unit circle or always stay outside. Hence, Corollary 1 establishes a crucial relationship between the fixed point stability properties and the mechanical dissipation properties as characterized by the loss term evaluated at the fixed point delay,  $\epsilon(\alpha^*)$ . The final piece of information relating the dissipation parameters,  $\{\mu, \zeta\}$ , to the first derivative of the remaining time function,  $\varphi_e'(\alpha^*)$  and, therefore, the stability properties of  $\mathbf{r}(\mathbf{p})$ , comes from Corollary 2.

**Corollary 2.** *The mechanical dissipation level, as characterized by the dissipation parameters,  $\{\mu, \zeta\}$ , determines the value of the first derivative of the remaining time function evaluated at the fixed point delay as follows:*

$$\begin{aligned} \mu \in (0, \infty) \wedge \zeta \in (0, 1) &\Rightarrow \varphi_e'(\alpha^*) \in (-1, 0) \\ \mu = 0 \wedge \zeta = 1 &\Rightarrow \varphi_e'(\alpha^*) = -1 \\ \mu \in (-\infty, 0) \wedge \zeta \in (1, \infty) &\Rightarrow \varphi_e'(\alpha^*) \in (-\infty, -1) \end{aligned} \tag{2.48}$$

PROOF. For the lossless case,  $\mu = 0$  and  $\zeta = 1$ , the loss term is unity,  $\epsilon(\alpha) \equiv 1$ , hence  $\varphi_e'(\alpha^*) = -1$ . Note that the loss term,  $\epsilon(\alpha)$ , is monotonically related to both viscous damping,  $\mu$ , and collision coefficient,  $\zeta$ ,

$$\begin{aligned}\frac{\partial\epsilon(\alpha)}{\partial\mu} &= -(\alpha + \varphi_e(\alpha) + \beta_{[1,e-1]})\epsilon(\alpha) < 0, \\ \frac{\partial\epsilon(\alpha)}{\partial\zeta} &= \frac{1}{\zeta}\epsilon(\alpha) > 0.\end{aligned}$$

Hence, for the lossy setting,  $\mu = (0, \infty)$  and  $\zeta = (0, 1)$ , the loss term is strictly less than unity,  $\epsilon(\alpha) < 1$ . Similarly, for the gainy case,  $\mu = (-\infty, 0)$  and  $\zeta = (1, \infty)$ , the loss term is strictly larger than unity,  $\epsilon(\alpha) > 1$ . The result in (2.48) follows directly from the relationship in (2.47).

□

The results of Corollary 1 and Corollary 2 establish a direct relationship between the dissipation parameters,  $\{\mu, \zeta\}$  and the locus of the eigenvalues,  $\lambda_{\pm}$ , for  $\mathbf{p}^* \in \mathcal{P}_+^*$  which ultimately leads to a dissipation based stability condition for the fixed points,  $\mathbf{p}^*$ , in  $\mathcal{P}_+^*$ . We will separately discuss the lossy case in Proposition 4 and the gainy case in Proposition 5. The lossless case will be covered by Lemma 8 and Conjecture 3.

**Proposition 4.** *Consider a lossy setting,  $\mu = (0, \infty)$  and  $\zeta = (0, 1)$ , and let  $\mathbf{p}^*$  be a fixed point in  $\mathcal{P}_+^*$ . If the clock period is in the “interval of stable periods,”*

$$T_c \in \mathcal{T}_s := (T_c^0, T_c^3), \tag{2.49}$$

*then the fixed point,  $\mathbf{p}^*$ , is asymptotically stable. Otherwise, the fixed point,  $\mathbf{p}^*$ , is unstable. The bounds of the interval of stable periods,  $T_c^0$  and  $T_c^3$ , are defined in (2.35) and (2.50), respectively.*

PROOF. It follows from Corollary 2 that for lossy settings the locus of the eigenvalues, as illustrated in Figure 2.11, are inside the unit circle for all  $T_c \in \mathcal{T}_{r1} \cup \mathcal{T}_c$ . For  $T_c \in \mathcal{T}_{r2}$  one of the eigenvalues,  $\lambda_+$ , always remains inside the unit circle whereas the other eigenvalue,  $\lambda_-$ , leaves the unit circle at

$$T_c^3 := \lambda_-^{-1}(-1). \quad (2.50)$$

We define  $\mathcal{T}_s := (T_c^0, T_c^3)$  as the “interval of stable periods” where the eigenvalues are inside the unit circle and the fixed point,  $\mathbf{p}^*$ , is asymptotically stable. Conversely, we define the complement set as the “interval of unstable periods,”  $\mathcal{T}_u := (T_c^3, \infty)$ , where  $\lambda_-$  is outside the unit circle and the fixed point,  $\mathbf{p}^*$ , is unstable.

□

**Proposition 5.** *Consider a gainy setting,  $\mu = (-\infty, 0)$  and  $\zeta = (1, \infty)$ , and let  $\mathbf{p}^*$  be a fixed point in  $\mathcal{P}_+^*$ . For all admissible clock periods,  $T_c \in \mathcal{T}$ , there is least one eigenvalue outside the unit circle, and therefore, the fixed point,  $\mathbf{p}^*$ , is unstable.*

PROOF. Corollary 2 states that for gainy mechanical hopper  $\varphi_e'(\alpha^*) \in (-\infty, -1)$ , and therefore, the locus of the eigenvalues, as illustrated in Figure 2.11, are outside the unit circle for all  $T_c \in \mathcal{T}$ , and therefore, all fixed points,  $\mathbf{p}^*$ , in  $\mathcal{P}_+^*$  are unstable. □

Finally, we will consider the lossless setting,  $\mu = 0$  and  $\zeta = 1$ . For unstable periods,  $T_c \in \mathcal{T}_u$ , fixed points,  $\mathbf{p}^*$ , in  $\mathcal{P}_+^*$ , are unstable as explained in Lemma 8.

**Lemma 8.** *Consider a lossless setting,  $\mu = 0$  and  $\zeta = 1$ , and let  $\mathbf{p}^*$  be a fixed point in  $\mathcal{P}_+^*$ . If the clock period is within the interval of unstable periods,  $T_c \in \mathcal{T}_u$ , then the fixed point,  $\mathbf{p}^*$ , is unstable.*

PROOF. The result follows from the derivation of the eigenvalues for  $\mathcal{P}_+^*$  class. If  $T_c \in \mathcal{T}_u$ , then one of the eigenvalues,  $\lambda_-$ , is guaranteed to be outside the unit circle, and therefore, the fixed point,  $\mathbf{p}^*$ , is unstable.  $\square$

Unfortunately, for the stable period interval,  $T_c \in \mathcal{T}_s$ , the linearization study cannot determine the stability properties of the lossless “full” nonlinear return map at a fixed point,  $\mathbf{p}^*$ , in  $\mathcal{P}_+^*$  since the eigenvalues of the Jacobian,  $\lambda_{\pm}$ , are located on the unit circle. If it could be shown that a Hopf bifurcation [120] occurs then this would imply that the nonlinear system is either asymptotically stable or unstable. However, our numerical studies, which will be presented in Section 2.5.1, suggest that the nonlinear system is stable in the sense of Lyapunov but is not attracting. That is, the full return map in (2.22) has quasi-periodic solutions arranged in concentric closed curves around the fixed point,  $\mathbf{p}^*$ . Conjecture 3 summarizes this stability observation. We will refer to this operating regime as “neutral stability.” The reader can refer to Section 2.5.2 for numerical bifurcation studies.

**Conjecture 3.** *Consider a lossless setting,  $\mu = 0$  and  $\zeta = 1$ , and let  $\mathbf{p}^*$  be a fixed point in  $\mathcal{P}_+^*$ . If the clock period is within the interval of stable period,  $T_c \in \mathcal{T}_s$ , then the fixed point,  $\mathbf{p}^*$ , is “neutrally stable,” that is, all solutions in a neighborhood are quasi-periodic orbits lying on closed curves that enclose it.*

## 2.5 Numerical Studies

### 2.5.1 Typical Stable Behaviors of the Coupled System

This section presents typical operating regimes of the coupled system and relates the properties of the Poincaré sequence,  $\mathbf{p}_k$ , to the motion of the mechanical hopper.

Our analysis in Section 2.4.4 was inconclusive for the lossless setting,  $\mu = 0$ . In Section 2.5.2 we will present a numerical bifurcation study that suggests that the lossless case gives rise to neutrally stable operating regimes. Figure 2.12 illustrates a typical run of a clock driven lossless mechanical hopper. The neutral stability of the fixed point,  $\mathbf{p}^*$ , implies that the apex height of the hops as well as the phase difference between the controller clock and the mechanical hopper will oscillate. The former is clearly depicted in Figure 2.12.

Theorem 1 states sufficient conditions that give rise to stable behaviors for lossy mechanical hoppers,  $\mu > 0$ . However, the analysis in Section 2.4.4 does not check for the existence of valid stable fixed points,  $\mathbf{p}^*$ . Our numerical studies suggest that for a wide range of physically relevant damping values,  $\mu$ , there is a sizable subset of shape configurations,  $\mathcal{O}_s(\mu) \subset \mathcal{O}$ , that results in valid stable fixed points. We will refer to this set as the “set of stable shapes.” Figure 2.13 presents a numerically computed set of stable shape configuration,  $\mathcal{O}_s$ , for two different damping levels. This particular numerical study concerns a 2-cell shape function,  $\psi(\theta)$ , that is parameterized by the duration of the first cell,  $\beta_1$ , and the difference between the two cell stiffnesses,  $\Delta\xi := \xi_1 - \xi_2$ .

Our analysis imposes no restriction on the number of cells, and the results of Section 2.4.4 apply to a wide range of excitation signals. Figure 2.14 and Figure 2.15 depicts two runs with 5-cell and 41-cell shape functions, respectively. This very feature of the analysis allows us to study the stability properties of arbitrary continuous excitation signals via piece-wise constant approximates.

## 2.5.2 Bifurcation Studies

This section will present a numerical parameter study of the return map,  $\mathbf{r}(\mathbf{p})$ . We will demonstrate how the limit properties of the fixed points,  $\mathbf{p}^*$ , vary with the

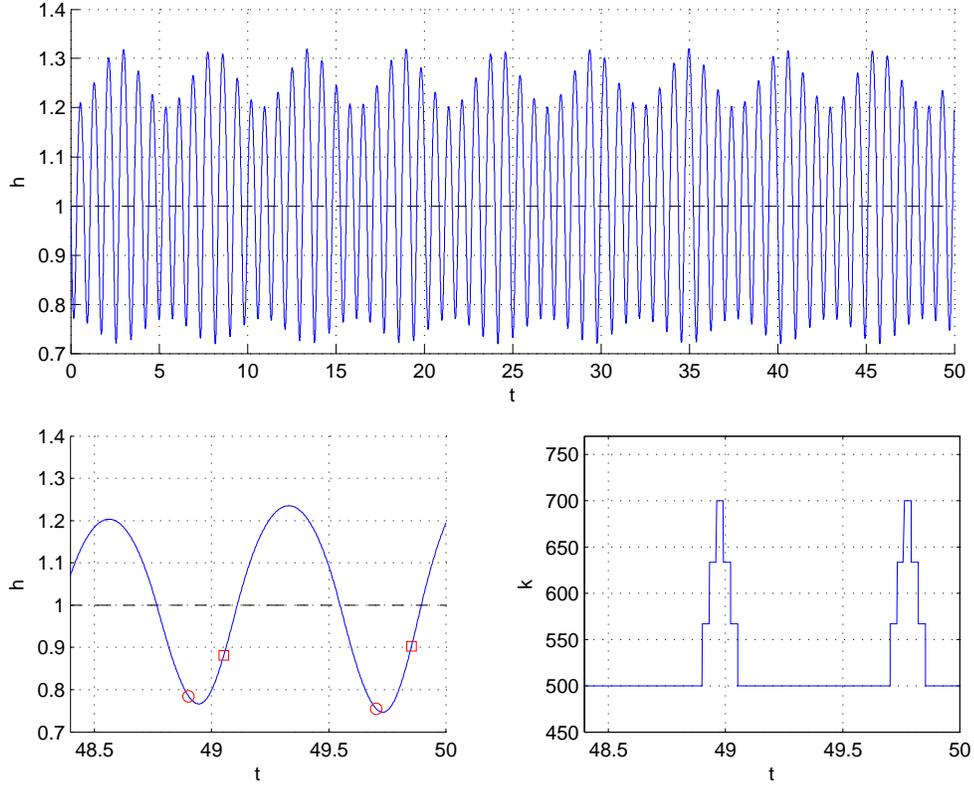


Figure 2.12: Typical evolution of the body height in a neutrally stable fundamental repeatable mode sequence,  $\mathcal{GAG}(6,6)$ . The mechanical system is lossless,  $\mu = 0$ , and the shape function,  $\psi(\theta)$ , takes a 6-cell triangular form. The top plot shows the body height across the entire experiment span. The bottom plots zoom in to the very last two cycles of the run. The bottom-left plot shows the body height. The circle markers indicate the event of clock reset and the square markers point out where triangle portion of the excitation shape concludes. Shape function,  $\psi(t)$ , is depicted in the bottom-left plot.

changes in certain key parameters. For this study we will focus on two parameters: 1) the clock period,  $T_c$ ; and 2) the mechanical damping coefficient,  $\mu$ .

The local stability analysis in Section 2.4.4 has already established the basic prop-

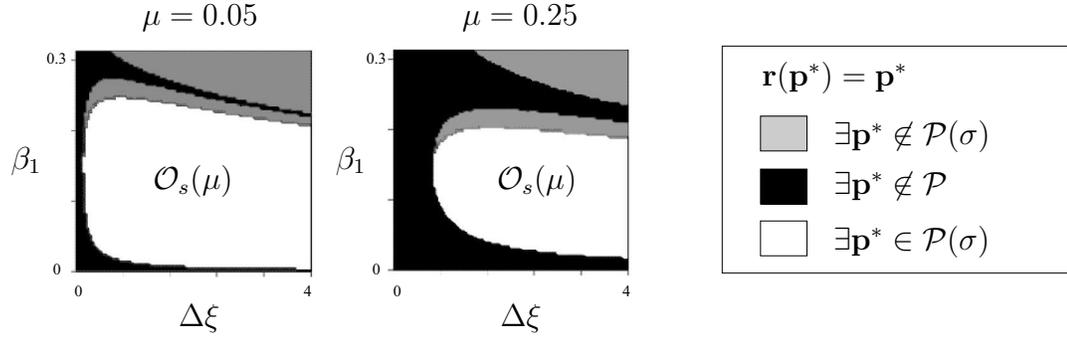


Figure 2.13: Numerically computed sets of stable shape configurations,  $\mathcal{O}_s(\mu)$ , for a coupled system driven by a 2-cell shape function,  $\psi(\theta)$ , operating in  $\mathcal{GAG}(2, 2)$ . The shape function is parameterized by the first cell duration,  $\beta_1$ , and the difference between the stiffness in the first and second cells,  $\Delta\xi := \xi_1 - \xi_2$ . The set of stable shape configurations,  $\mathcal{O}_s(\mu)$ , is a function of the damping coefficient,  $\mu$ . Two different damping levels depict the relationship between mechanical losses and actuation.

erties of the eigenvalues' loci as a function of the clock period,  $T_c$ . The reader can refer to Figure 2.16 for a set of plots that demonstrate the relationship between the eigenvalues,  $\lambda_{\pm}$  and the clock period,  $T_c$ . Recall that the clock period,  $T_c$ , does not appear in the forcing function expression,  $\rho(\alpha)$ , and therefore, the existence of fixed points and their delay states,  $\alpha^*$ , are not affected by the variations in the clock period,  $T_c$ .

For this discussion we limit our attention to the physically relevant admissible clock periods,  $T_c > T_c^0$ , where  $T_c^0$  is the smallest admissible clock period defined in (2.35). We will consider a lossy mechanical system,  $\mu > 0$  and  $\zeta \in (0, 1)$ , and a properly selected shape configuration,  $\mathbf{o}$ , such that the forcing function,  $\rho(\alpha)$ , crosses unity (Figure 2.16(top)) and defines two fixed points of different types:  $\mathbf{p}_-^* \in \mathcal{P}_-$ , and  $\mathbf{p}_+^* \in \mathcal{P}_+$ .

The eigenvalues of  $\mathbf{p}_+^*$  are located inside the unit circle over a bounded period

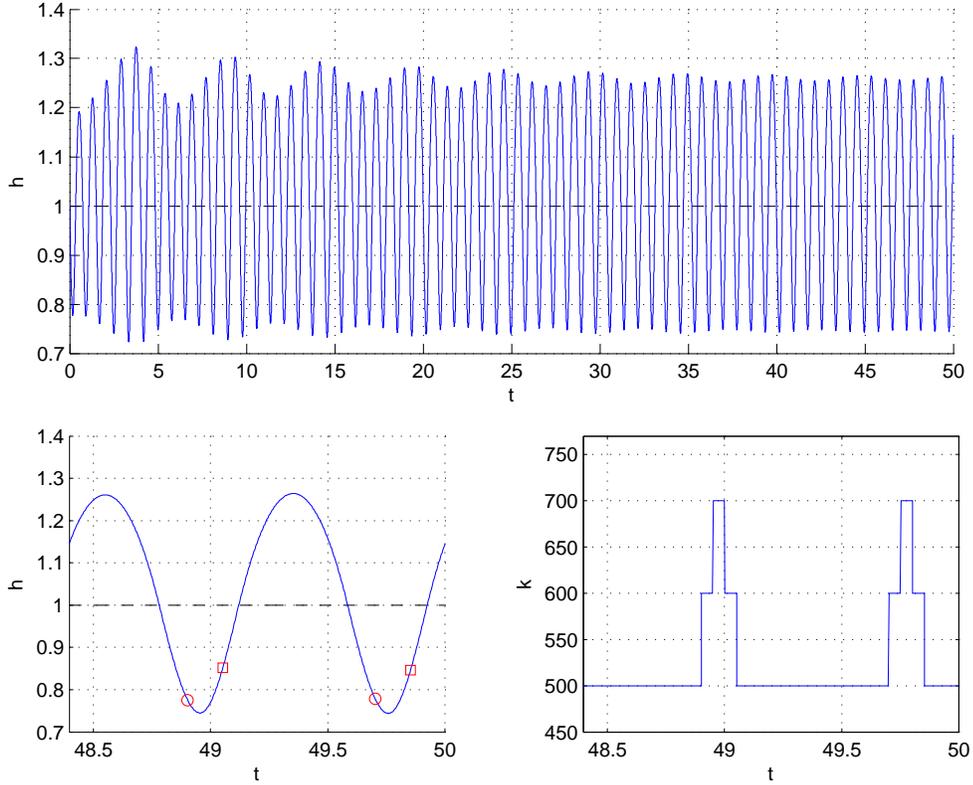


Figure 2.14: Typical evolution of the body height in a stable fundamental repeatable mode sequence,  $\mathcal{GAG}(4, 5)$ . In these experiments the mechanical system is lossy and the shape function,  $\psi(\theta)$ , takes a 4-cell piece-wise constant approximation of a triangle form. The top plots show the body height across the entire experiment span. The two plots in the bottom zoom in to the very last two cycles of the run. The bottom-left plot shows the limit mechanical behavior. The circle markers indicate the event of clock reset,  $\theta = 0$ , and the square markers point out where triangle portion of the excitation shape concludes,  $\theta = \theta_4$ . The bottom-right plot depicts the shape function,  $\psi(t)$ , during the last two periods of clock.

interval,  $T_c \in \mathcal{T}_s = (T_c^0, T_c^3)$ , where  $\mathcal{T}_s$  is the interval of stable periods defined in (2.49). At  $T_c = T_c^3$  a saddle node bifurcation occurs and the fixed point,  $\mathbf{p}_+$ , becomes

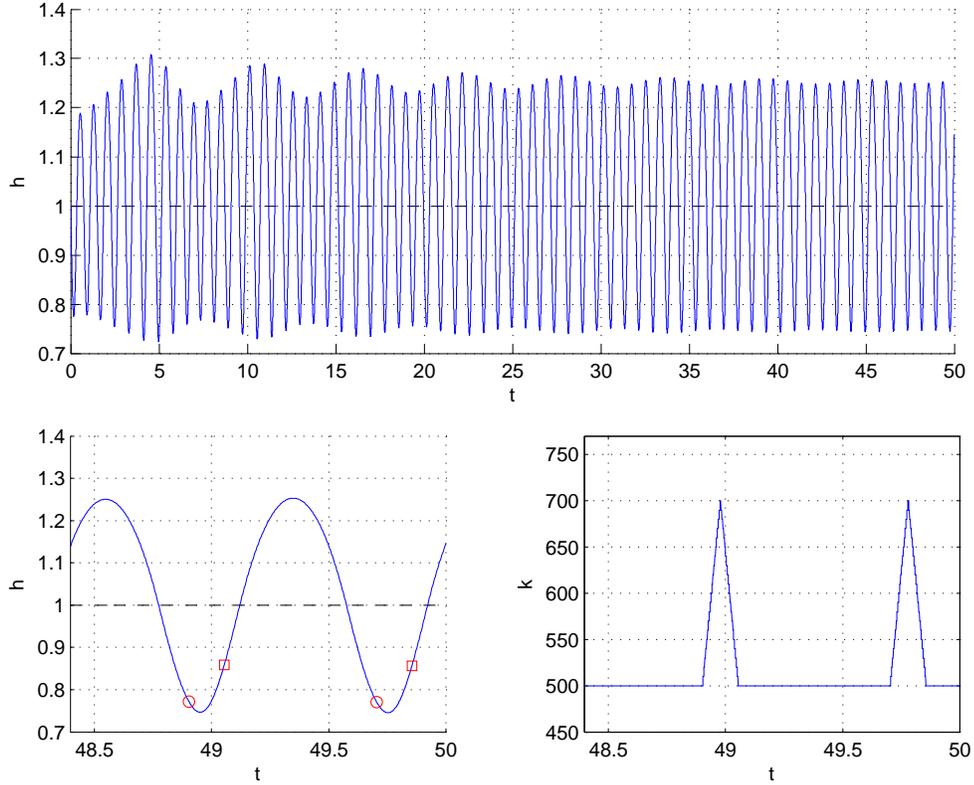


Figure 2.15: Typical evolution of the body height in a stable fundamental repeatable mode sequence,  $\mathcal{GAG}(40, 41)$ . In these experiments the mechanical system is lossy and the shape function,  $\psi(\theta)$ , takes a 41-cell piece-wise constant approximation of a triangle form. The top plots show the body height across the entire experiment span. The two plots in the bottom zoom in to the very last two periods of the run. The bottom-left plot shows the limit mechanical behavior. The circle markers indicate the event of clock reset,  $\theta = 0$ , and the square markers point out where triangle portion of the excitation shape concludes,  $\theta = \theta_{40}$ . The bottom-right plot depicts the shape function,  $\psi(t)$ , during the last two clock periods.

unstable. On the other hand,  $\mathbf{p}_*$  is an unstable fixed point for all clock period values since one of its eigenvalue,  $\lambda_+$ , is strictly outside the unit circle for all admissible

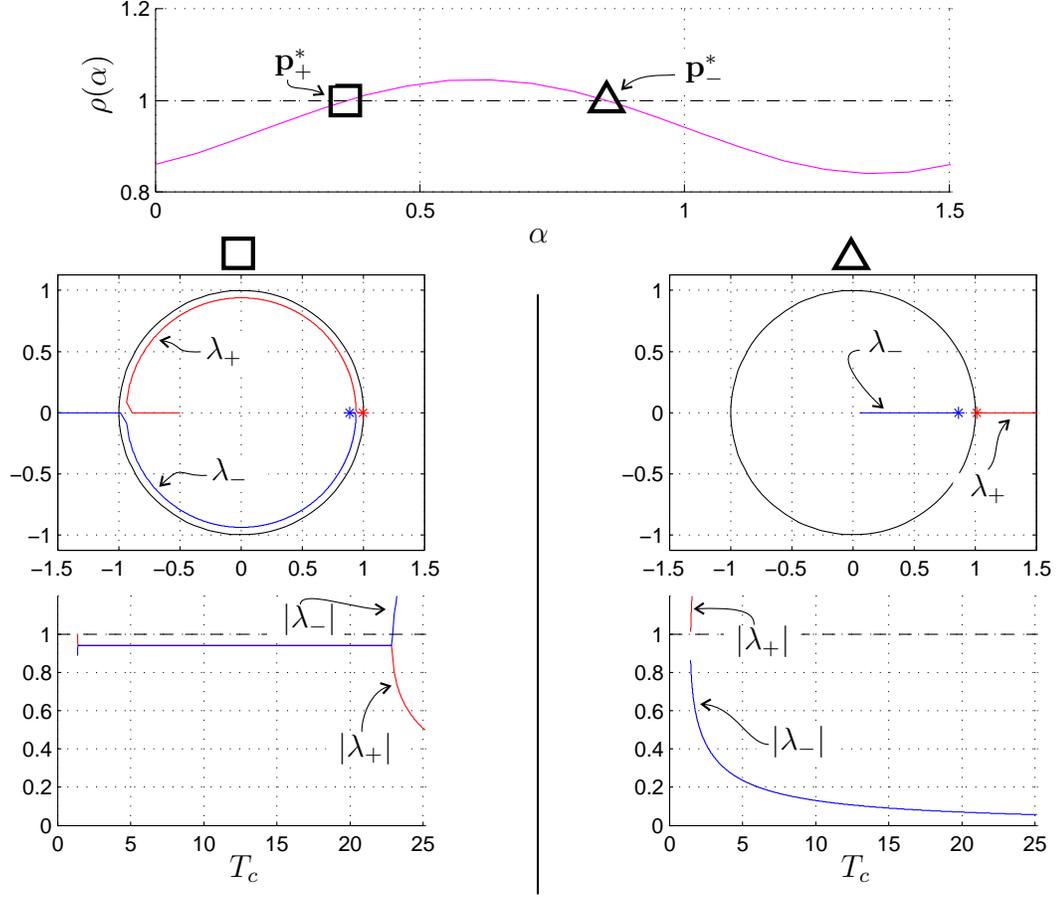


Figure 2.16: Numerically computed locus of eigenvalues as the clock period,  $T_c$ , varies. The top plot depicts the forcing function,  $\rho(\alpha)$ , at the chosen shape configuration,  $\mathbf{o}$ , whose two unity crossings in the delay interval,  $\mathcal{D}$ , define two fixed points,  $\mathbf{p}_-^* \in \mathcal{P}_-$ , and  $\mathbf{p}_+^* \in \mathcal{P}_+$ . The middle plots show the loci of the two eigenvalues,  $\lambda_-$  (blue) and  $\lambda_+$  (red), for each fixed point: left is for  $\mathbf{p}_+^*$ ; right is for  $\mathbf{p}_-^*$ . The asterisks indicate the eigenvalues at the smallest admissible clock period,  $T_c = T_c^0$ . The bottom plots depict the magnitudes of the eigenvalues,  $\lambda_{\pm}$ , as a function of the clock period,  $T_c$ .

clock periods,  $T_c \in \mathcal{T}$ .

Another parameter that has an important role in shaping the limit properties of the coupled system is the damping coefficient,  $\mu$ , which determines the dissipation

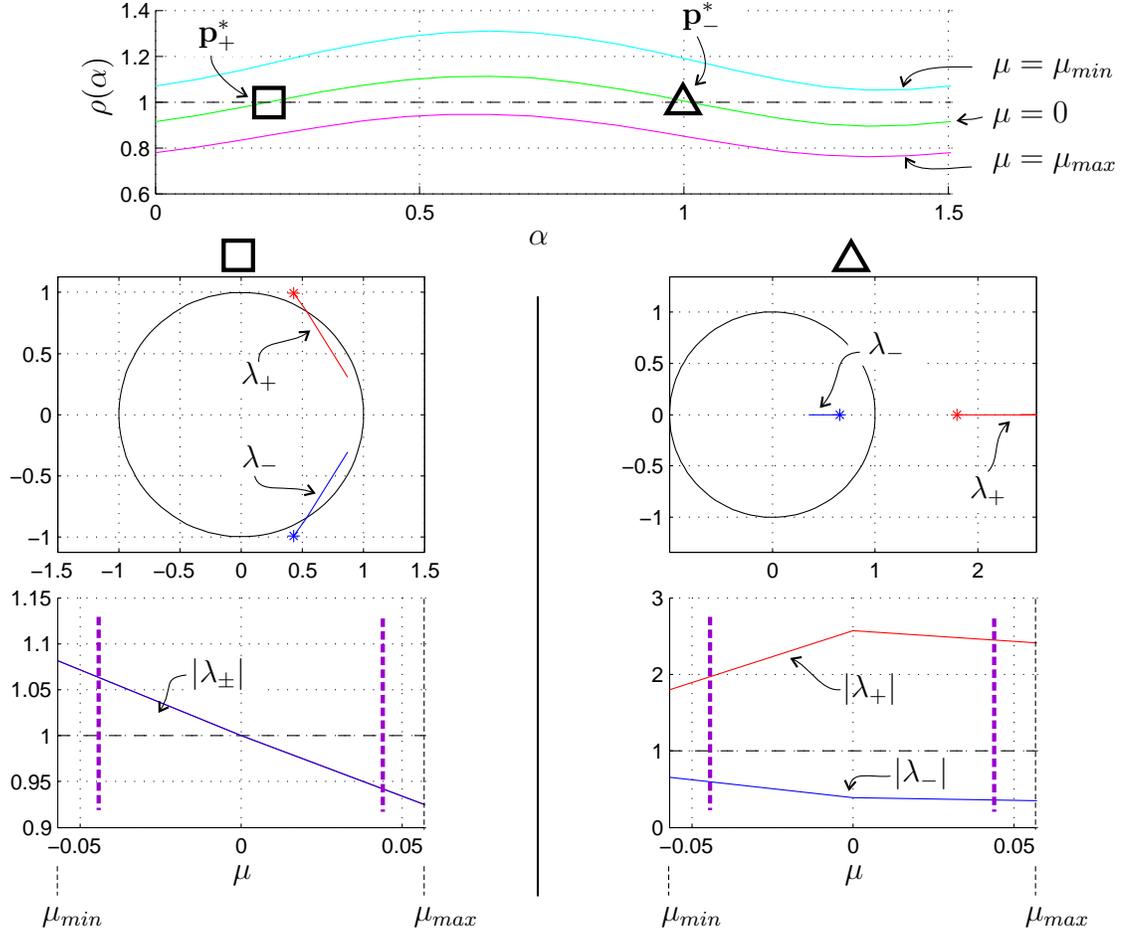


Figure 2.17: Numerically computed locus of eigenvalues as the damping coefficient,  $\mu$ , varies. The shape configuration,  $\mathbf{o}$ , and the clock period,  $T_c$ , are kept constant. The damping coefficient spans an interval,  $(\mu_-, \mu_+)$ , which contains the lossless case,  $\mu = 0$ . The top plot shows the forcing function,  $\rho(\alpha)$ , for gainy,  $\mu = \mu_-$ , lossless,  $\mu = 0$ , and lossy,  $\mu = \mu_+$ , cases. We consider only those damping values that give rise to fixed points,  $\mathbf{p}_\pm^*$ . The middle plots show the loci of the two eigenvalues,  $\lambda_-$  (blue) and  $\lambda_+$  (red), for each fixed point: left is for  $\mathbf{p}_+^*$ ; right is for  $\mathbf{p}_-^*$ . The asterisks indicate the eigenvalues at  $\mu = \mu_-$ . The bottom plots depict the magnitudes of the eigenvalues,  $\lambda_{pm}$ , as a function of the damping,  $\mu$ .

characteristics of the mechanical system. For our immediate discussion we will focus on only those fixed points,  $\mathbf{p}^*$ , that are in the  $\mathcal{P}_+^*$  set, and ignore all others, which they are guaranteed to be unstable independent of the damping coefficient,  $\mu$ , and clock period,  $T_c$ , according to Proposition 3 and Lemma 4. Reader can refer to Figure 2.17 for plots of eigenvalue loci as a function of damping coefficient,  $\mu$ .

Consider a fixed point  $\mathbf{p}^* \in \mathcal{P}_+^*$  and let  $\lambda_{\pm}$  be the eigenvalues of the Jacobian evaluated at this point,  $\mathbf{J}(\mathbf{p}^*)$ . In Section 2.4.4 we established that the loci of the eigenvalues is parameterized by the first derivative of the remaining time function evaluated at the fixed point delay,  $\varphi_e'(\alpha^*)$ , which is related to the loss term evaluated at the same point,  $\epsilon(\alpha^*)$ , according to Corollary 1. The loss term,  $\epsilon(\alpha)$ , is a monotonic function of the damping,  $\mu$ , which allows us to assess the changes in the coupled behavior as the damping coefficient varies.

For lossy settings,  $\mu > 0$ , the eigenvalues are located inside the unit circle for all stable clock periods,  $T_c \in \mathcal{T}_s$ . Similarly, one can conclude that for gainy settings,  $\mu < 0$ , the eigenvalues are outside the unit circle and the coupled system is unstable.

In the lossless case,  $\mu = 0$ , the loss term is identical to unity,  $\epsilon(\alpha) \equiv 1$ , and therefore, the eigenvalues lie on the unit circle. Typically, we cannot conclude stability properties of the nonlinear system from its linearization in the lossless case. In numerical bifurcation studies we observed that when the damping coefficient vanishes,  $\mu = 0$ , a Hopf bifurcation [120] occurs suggesting that the fixed points,  $\mathbf{p}^* \in \mathcal{P}^*$ , are neutrally stable in the lossless case,  $\mu = 0$ . This claim is supported by numerical studies on the original system which is summarized in Figure 2.18.

### 2.5.3 Numerical Assessment of the Stability Properties

This section will present a set of numerical studies aiming to characterize the stability properties of the “full” nonlinear return map,  $\mathbf{r}(\mathbf{p})$ , in (2.22) in relation to

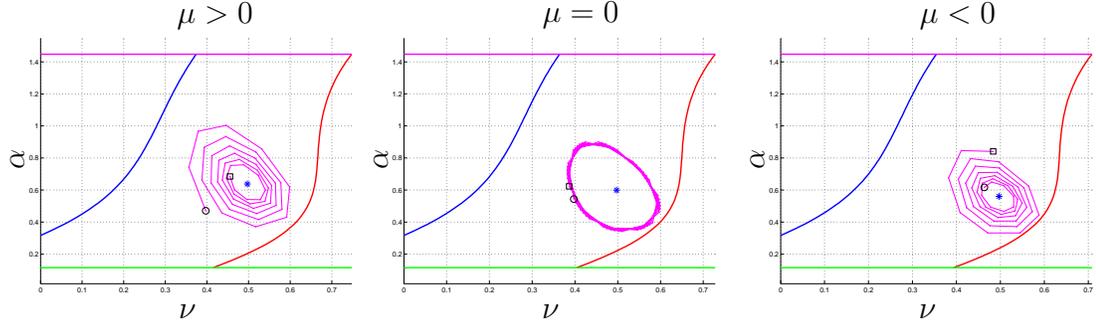


Figure 2.18: The discrete flow of the Poincaré states for three different mechanical dissipation settings: lossy,  $\mu > 0$ ; lossless,  $\mu = 0$ ; and gainy,  $\mu < 0$ . The boundaries of the valid mode domain,  $\mathcal{P}(\sigma)$ , are indicated in each plot. The (blue) star indicates the (only) valid fixed point. The initial and final conditions of each flow are marked by circles and squares, respectively. The lossy setting is locally asymptotically stable. The lossless case demonstrates quasi-periodic neutrally stable behavior. Finally, the gainy setup is unstable.

the system parameters. As in the bifurcation study in Section 2.5.2, we will primarily focus on two system parameters: the damping coefficient,  $\mu$ , which characterizes the mechanical dissipation level; and the clock period,  $T_c$ .

To assess the stability properties of the nonlinear return map,  $\mathbf{r}(\mathbf{p})$ , we will employ two (numerically computed) measures: the convergence speed; and the size of the basin. In this context, convergence speed will refer to the time constant of the envelope of convergence for the 2-dimensional return map,  $\mathbf{r}(\mathbf{p})$ . Below we will first provide detailed definitions of these two measures. Next, we will discuss the variations of these measures as the damping coefficient,  $\mu$ , and clock period,  $T_c$ , varies.

For our numerical study we define a surrogate convergence speed,

$$\text{convergence} := \frac{1}{K} (\|\mathbf{p}_1 - \mathbf{p}_0\|_2 - \|\mathbf{p}_K - \mathbf{p}_{K-1}\|_2), \quad (2.51)$$

where  $K$  is the length of a given sequence and  $\mathbf{p}_0$  is the initial condition. The faster a sequence converges to a limit the larger is the convergence speed in (2.51).

We employ a slightly modified definition of the basin of attraction which we will denote by  $\overline{\mathcal{P}}(\sigma)$ . Recall that the return map,  $\mathbf{r}(\mathbf{p})$ , of a mode sequence,  $\sigma$ , is only physically relevant over the corresponding valid domain,  $\mathcal{P}(\sigma)$ . Let  $\{\mathbf{p}_k\}_{k=0}^{\infty}$  be the sequence generated by consecutive iterations of the return map,  $\mathbf{r}(\mathbf{p})$ , starting from an initial condition,  $\mathbf{p}_0$ . We conclude that the initial condition,  $\mathbf{p}_0$ , is in the basin of the valid fixed point,  $\mathbf{p}^*$ , if and only if, the sequence converges to the fixed point,  $\lim_{k \rightarrow \infty} [\mathbf{p}_k] = \mathbf{p}^*$ , and the sequence,  $\{\mathbf{p}_k\}_{k=0}^{\infty}$ , remains in the valid domain at all times,  $\forall k, \mathbf{p}_k \in \mathcal{P}(\sigma)$ . That is, by definition, the basin of a fixed point,  $\mathbf{p}^*$ , is restricted to be within the valid domain,  $\mathcal{P}(\sigma)$ .

It follows from the results from Section 2.4.4 that the damping,  $\mu$ , is inversely proportional to the time constant of the local convergence envelope. Recall that the loss term evaluated at the fixed point delay,  $\epsilon(\alpha^*)$ , determines the magnitude of the eigenvalues,  $\lambda_{\pm}$ , according to the derivation in (2.44) and the results in Corollary 1. Since the loss term,  $\epsilon(\alpha)$ , is a strictly decreasing function of the damping,  $\mu$ , as damping increases the eigenvalues,  $\lambda_{\pm}$ , of the stable fixed point,  $\mathbf{p}^*$ , get closer to the origin, and therefore, the time constant of the local convergence envelope gets faster. Figure 2.19 illustrates iterations of the return map,  $\mathbf{r}(\mathbf{p})$ , for different damping values,  $\mu$ . The reader can find the corresponding list of convergence speeds in Table 2.1.

Plot in Figure 2.19	a	b	c	d	e	f	g	h
$\mu$	0	0.01	0.02	0.03	0.04	0.05	0.06	0.07
convergence	0.0	0.04	0.07	0.10	0.12	0.15	0.18	0.23

Table 2.1: The convergence speed measured for the runs presented in Figure 2.19.

Figure 2.20 illustrates a number of numerically computed basins at various damp-

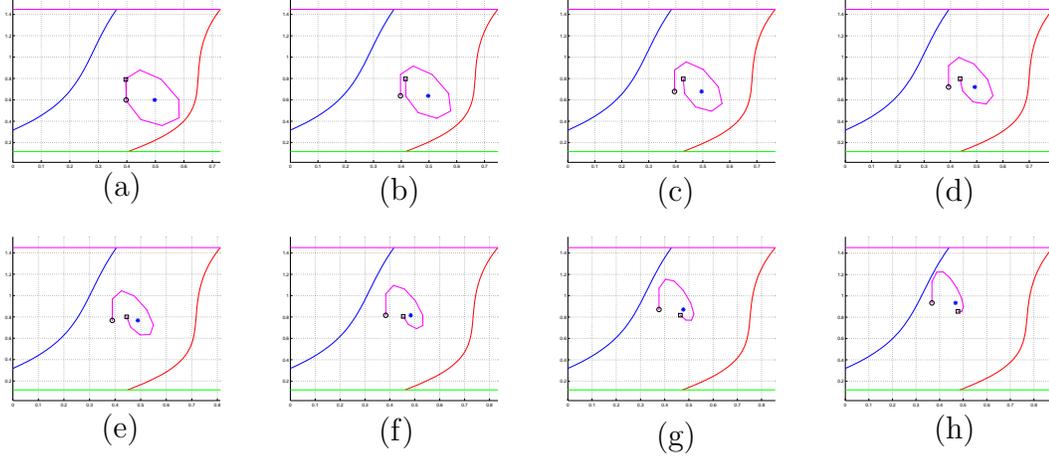


Figure 2.19: The discrete Poincaré map at eight different mechanical dissipation levels: (a)  $\mu = 0$ ; (b)  $\mu = 0.01$ ; (c)  $\mu = 0.02$ ; (d)  $\mu = 0.03$ ; (e)  $\mu = 0.04$ ; (f)  $\mu = 0.05$ ; (g)  $\mu = 0.06$ ; and (h)  $\mu = 0.07$ . In each plot the valid mode sequence domain,  $\mathcal{P}(\sigma)$ , is indicated. The initial condition of each experiment is displaced from its stable fixed point by a constant distance. The return map is iterated 10 times. The convergence speed is measure by the surrogate in (2.51). The convergence speed for each run can be found in Table 2.1.

ing levels. The coupled system is neutrally stable in the lossless setting,  $\mu = 0$ . The invariant set for the lossless setting is depicted in a different color. The relationship between the damping,  $\mu$ , and the size of the basin is not monotonic. Our numerical results suggests that there exists an optimal damping for a given controller configuration,  $(\mathbf{o}, T_c)$ , where the basin size is maximal.

The cases presented in Figure 2.19 and Figure 2.20 have the same parameters. Note that there is an optimal damping value where the basin is the largest. However, for faster convergence the damping needs to be increased further which gives rise to a trade-off between the speed of convergence and the robustness against external disturbances.

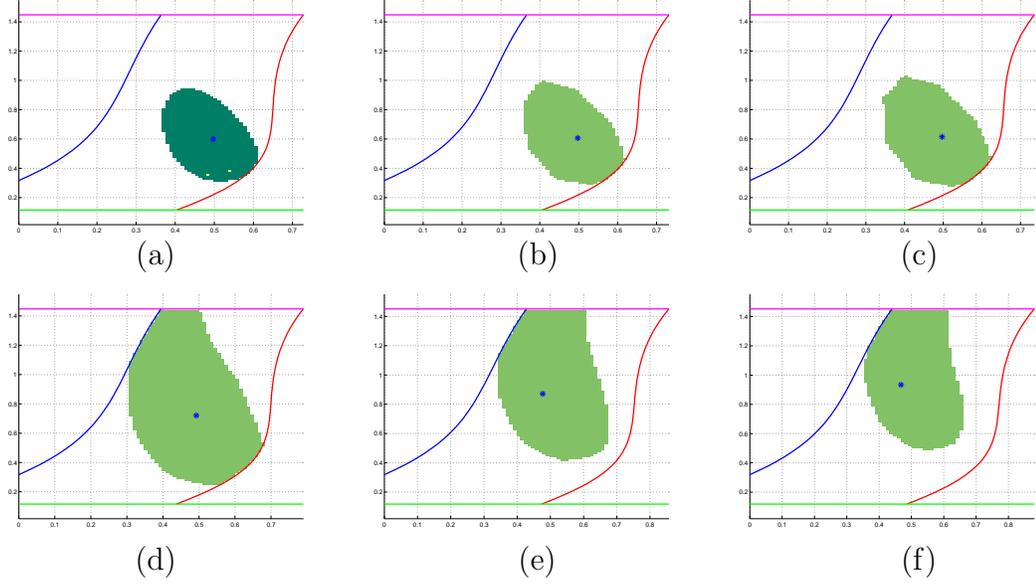


Figure 2.20: The basin of the stable fixed point,  $\mathbf{p}_+^*$ , for six different damping levels: (a)  $\mu = 0$ ; (b)  $\mu = 0.002$ ; (c)  $\mu = 0.004$ ; (d)  $\mu = 0.01$ ; (e)  $\mu = 0.03$ ; and (f)  $\mu = 0.07$ . In all plots except the lossless case in (a) the color coded area within the valid cell,  $\mathcal{P}(\sigma)$ , is the numerically computed domain of attraction of the stable fixed point which is indicated by a (blue) star. In the lossless case in (a) the color coded area is the invariant set under the return map,  $\mathbf{r}(\mathbf{p})$ . However, the discrete map does not converge to the fixed point,  $\mathbf{p}^*$ .

The local stability analysis in Section 2.4.4 concluded that the open-loop clock controller can lead to asymptotically stable behaviors only if the clock period,  $T_c$ , is chosen from the stable period interval,  $\mathcal{T}_s = (T_c^0, T_c^3)$ . Figure 2.21 depicts the variation in the basin of attraction as the clock period,  $T_c$ , sweeps the stable period interval,  $\mathcal{T}_s$ . The basin vanishes as the clock period,  $T_c^3$ , reaches to the upper bound of the stable periods,  $T_c^3$ , where the return map,  $\mathbf{r}(\mathbf{p})$ , goes through a saddle node bifurcation as demonstrated in Section 2.5.2.

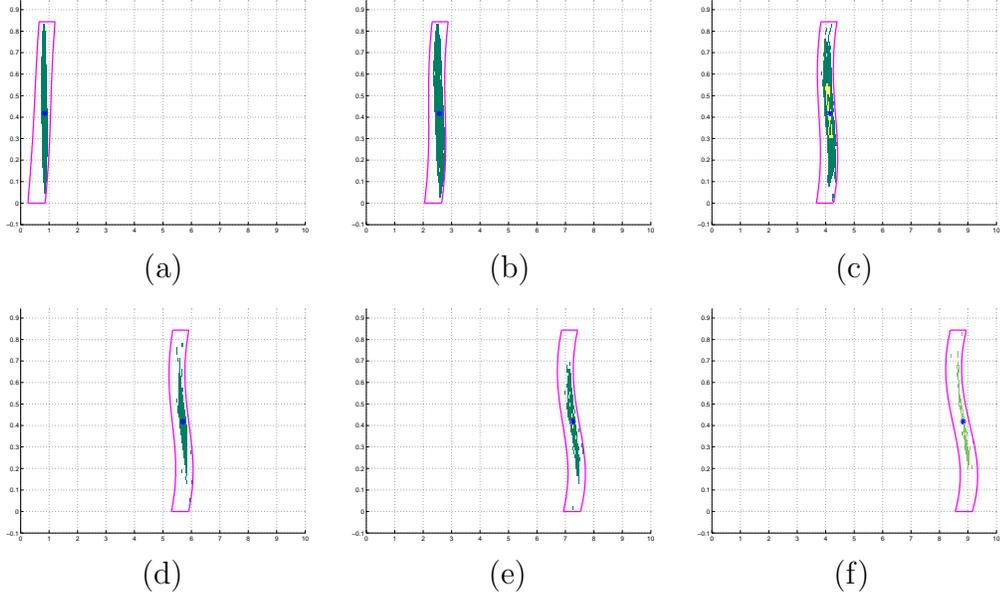


Figure 2.21: The basin of the stable fixed point,  $\mathbf{p}_+^*$ , for six different clock periods: (a)  $T_c = 0.9$ ; (b)  $T_c = 2.0$ ; (c)  $T_c = 3.0$ ; (d)  $T_c = 4.0$ ; (e)  $T_c = 5.0$ ; and (f)  $T_c = 6.0$ . The color coded area within the valid cell,  $\mathcal{P}(\sigma)$ , is the numerically computed domain of attraction of the stable fixed point which is indicated by a (blue) star.

## 2.6 Extensions of the Analysis

### 2.6.1 Open-Loop Controller Design

The analysis in Section 2.4.4 paves the way toward an open-loop controller design framework. The reader can find a detailed discussion of our preliminary results on open-loop controller design in Appendix B. This section will summarize those results.

Our numerical studies suggest that sufficient conditions for stability in Theorem 1 are satisfied by all stable configurations, and therefore, these conditions can effectively serve as guidelines for open-loop controller design. According to Theorem 1 the functional properties of the forcing function,  $\rho(\alpha)$ , and the remaining time function,  $\varphi_e(\alpha)$ , are the determining factors in the stability of the coupled system. However,

the relationship between the shape configuration,  $\mathbf{o}$ , and these key terms of the return map is too complex to lend itself to a simple design process.

We observe that, limited to a physically relevant family of operating regimes, which we call “small action regimes,” both the forcing function,  $\rho(\alpha)$ , and the remaining time function,  $\varphi_e(\alpha)$ , can be approximated by considerably simpler functions. Using these approximations instead of the original functions leads to an algorithmic process that returns clock parameters, the clock period,  $T_c$ , and the shape configuration,  $\mathbf{o}$ , that approximately yield a user-specified hopping task. We demonstrate this process in a case study.

Moreover, this design discussion offers some preliminary insight on the relationship between the form of the open-loop excitation signal, which is defined by the shape function,  $\psi(\theta)$ , and stability properties. We divide a typical shape function,  $\psi(\theta)$ , into two parts: an active part, where the stiffness changes; and an inactive part, where the stiffness remains constant. The former, whose duration is a constant and denoted by  $\beta_{[1,e-1]}$ , coincides with the stance mode,  $\mathcal{G}$ , and causes changes in the mechanical energy. We observe that the magnitude of the change in total mechanical energy monotonically increases with the duration of the active part,  $\beta_{[1,e-1]}$ , as well as the magnitude of the stiffness variations. Note that this result is analogous to the special case presented in Figure 2.13.

## 2.6.2 The Role of Mechanical Properties

The passive dynamics of the underlying mechanical system plays an important part in any open-loop control system. Naturally, there are “favorable” mechanical dynamics that give rise to better behavioral stability, that is, larger basins and faster convergence. The linear hopper in Section 2.1.1 is an illustrative model that offers analytical tractability. Yet, it is not a good representation of the physical world. In

Appendix C we present a numerical study that initiates an inquiry into the mechanical aspects of the open-loop control in a 1-DOF setting. This section will summarize some of these observations.

Our analysis of the linear prismatic hopper in Section 2.4 indicates that stabilization in the clock driven 1-DOF hopper is the result of two properties of the mechanical dynamics working together. We will briefly explain how these properties effectively relate the mechanical oscillation period,  $T_m$ , which encodes the hopping task, to the phase difference between the controller clock and the mechanical clock, the hopper. First, the clock-dictated modulations of leg stiffness,  $\xi$ , result in a change in the total mechanical energy,  $H$ , the amount of change being a function of the phase difference between the controller clock and the mechanical clock. In the clock driven 1-DOF linear hopper model this feature is captured by the forcing function,  $\rho(\alpha)$ . Second, the period of the mechanical oscillation,  $T_m$ , is monotonically related to the total mechanical energy,  $H$ . In the 1-DOF linear hopper the period of hopping increases with increasing energy by the virtue of the gravitational potential (airborne times increase with total mechanical energy). These two properties of the mechanical system effectively use the total mechanical energy as an intermediate to relate the phase difference between the clock controller and the mechanical clock to the period of the mechanical oscillations. This process underlies the success of the open-loop control.

Inspired by this observation we ask if certain types of spring laws offer better stability properties in open-loop control settings. Our numerical studies suggest that we can categorize spring laws according to their period-energy relationships which divides the spring laws into three classes: softening springs; linear springs; and hardening springs. Our preliminary results suggest that certain types of springs are suitable for certain types of tasks. For instance, softening springs offer a better solution for hopping tasks in which the spring goes through a compression followed by a de-

compression during stance. For a hopper we numerically demonstrate that the basin of attraction shrinks if the leg spring is of hardening type. On the other hand, a hardening spring is better suited for a climbing task where the leg spring typically goes through a decompression followed by a compression.

This study, although preliminary and limited in its scope, offers the beginnings of a set of guidelines for mechanical design to achieve robust open-loop control to accomplish rhythmic tasks.

## CHAPTER 3

# Feedback Adjustment of the Clock Driven 1-DOF Hopper

The open-loop controlled 1-DOF hopper exhibits poor stability properties. Our numerical studies in Section 2.5.3 demonstrated that the forward coupled system in Section 2.1.3 suffers from a small basin of attraction and slow convergence to the limit. Naturally, these shortcomings of the open-loop control make it unsuitable for practical application where the modeling inaccuracies and numerous sources of external disturbance easily destabilize the behavior.

The fundamental reason behind the poor stability features of the open-loop control is the lack of task-level information which leads to inefficient and sometimes incorrect excitation of the underlying mechanical system. Naturally, informing the controller about the state of task can lead to better control of the mechanical behavior. Previous work on open-loop control also identified these shortcomings and proposed discrete-time [126, 128, 129] and/or continuous-time [7, 55] feedback mechanisms to augment the open-loop controller in order to improve the robustness. However, none of these studies provide a rigorous formal discussion on why and how these feedback controllers work. This chapter will offer a careful formal discussion on how to introduce feedback

into the clock controllers.

We will frequently refer to the analysis of the forward coupled system in Chapter 2. In order to avoid any confusion between these two sets of terms/expressions our explanations will adopt a naming convention where all terms/expressions that are associated with a forward coupled setting will be tagged by “open-loop.” Similarly, all terms/expressions that are related to a feedback system will be tagged by “closed-loop.”

In this discussion we will consider a special family of feedback controllers. A feedback controller consists of a clock controller and a “tuning mechanism” that periodically *tunes* the clock controller according to the states of the coupled system. We will present a Poincaré analysis of the (generic) closed-loop setting in Section 3.2. We will present two instances of the family of feedback controllers whose generic structure is defined in Section 3.1. In each case we will demonstrate that the “proper” discrete-time corrections to the clock period,  $T_c$ , can lead to drastic improvements in the robustness of the limit behaviors.

Section 3.3 introduces a family of tuning mechanisms that improve the transients. Proposition 6 states sufficient conditions for the time constant of the envelope of the (underdamped) linearized response of the closed-loop system to get faster. These tuning laws correct the controller clock period,  $T_c$ , as a function of the delay state,  $\alpha$  — a surrogate for the phase difference between the clock and the mechanical hopper. It directly follows from Proposition 6 that the lossless case,  $\mu = 0$  and  $\zeta \in (0, 1)$ , which is only neutrally stable in the open-loop case, can be rendered asymptotically stable in a closed-loop system.

In Section 3.4, we define another tuning mechanism that effectively enlarges the basin of a fixed point. We achieve this by sequentially composing the basins of a family of operating regimes that are parameterized by the clock period,  $T_c$ . Proposition 7

states that under certain conditions the valid domain spans the entire speed axis. Our numerical results suggest that the basin of attraction extends across all stable speeds — those speed values given by (2.33) where the clock period,  $T_c$ , is limited to the the stable clock periods,  $\mathcal{T}_s$ . In Section 3.5, by appropriately combining the two tuning laws from Section 3.2.2 and Section 3.4 we construct a hierarchical tuning mechanism that combines the advantages attending its individual layers.

### 3.1 Feedback Controller

Numerous properties of the open-loop controlled 1-DOF hopper will serve as the foundation of this feedback discussion. In order to take advantage of the results from Chapter 2, we will limit our focus to a special family of feedback controllers where the structure of closed-loop settings remains similar to that of the open-loop settings.

The feedback controllers of interest (depicted in Figure 3.1) consist of two parts: 1) a clock controller; and 2) a “tuning mechanism.” The clock controller, described in Section 2.1.2, is a *tunable* system whose parameters, the clock period,  $T_c$ , and the shape configuration,  $\mathbf{o}$ , can be freely chosen. The tuning mechanism is a discrete-time feedback law that sits on top of this clock controller updating its parameters,  $\{T_c, \mathbf{o}\}$ , as a function of the Poincaré states,  $(\nu_k, \alpha_k)$ , that are sampled at the event of clock reset,  $\theta = 0$ . Clock parameters,  $\{T_c, \mathbf{o}\}$ , remain constant from one clock reset to the next.

The feedback controller design is driven by the properties of the open-loop setting. We refer to those clock cells that partially or completely coincide with the stance mode,  $\Theta_i$ ,  $i = 1, 2, \dots, e, N$ , as the “stance cells.” The open-loop return map,  $\mathbf{r}(\mathbf{p})$ , in (2.22) is parameterized by the clock period,  $T_c$ , and the durations of the stance cells,  $\beta_i$   $i = 1, 2, \dots, e, N$ . The special analytic structure of the open-loop return map,

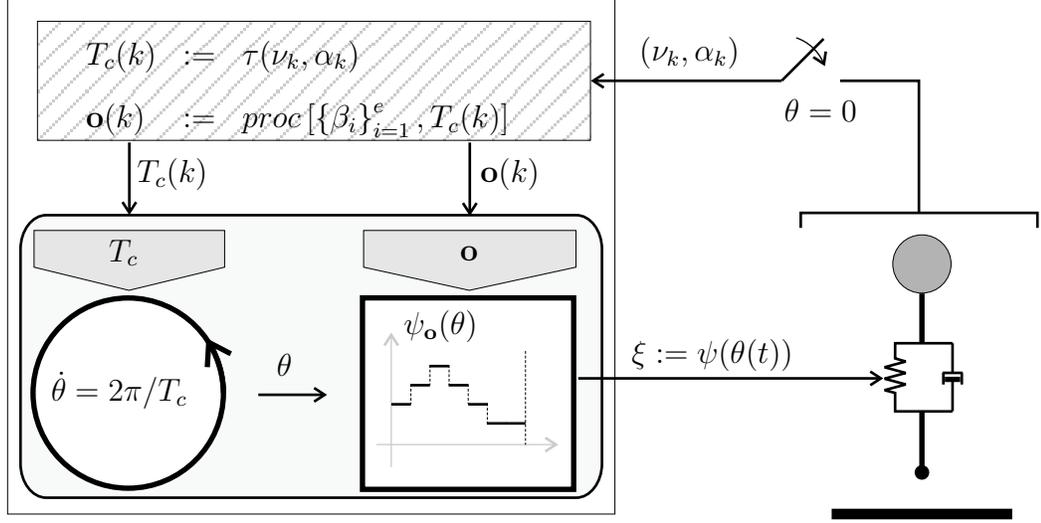


Figure 3.1: The generic structure of the closed loop system, consisting of a clock controller and a tuning mechanism. The states of the coupled system,  $(\nu, \alpha)$ , are sampled at the event of clock reset,  $\theta = 0$ . The tuning mechanism updates the clock parameters, the clock period,  $T_c$ , and the shape configuration,  $\mathbf{o}$ , as described by a tuning law — a functional relationship between the Poincaré states,  $(\nu, \alpha)$ , and the clock period,  $T_c$ . The adjustments to the shape,  $\mathbf{o}$ , are performed such that the durations of the stance cells,  $\beta_i$ ,  $i = 1, \dots, e, N$ , are kept constant. (Note that  $e + 1 = N$  is not necessarily true.)

$\mathbf{r}(\mathbf{p})$ , suggests that one can significantly improve the robustness of the limit behavior (faster convergence and larger basin) by “properly” adjusting the clock period,  $T_c$ . Inspired by this observation we choose to specify a tuning mechanisms by defining its “tuning law” — a functional relationship between the clock period,  $T_c$ , and the Poincaré states,  $(\nu, \alpha)$ . Section 3.1.1 will introduce a family of tuning laws that we will consider in this discussion. Section 3.1.2 describes a procedure that adjusts the shape configuration,  $\mathbf{o}$ , to keep the stance cell durations,  $\beta_i$   $i = 1, 2, \dots, e, N$ , constant as the clock period,  $T_c$ , changes.

### 3.1.1 A Family of Tuning Laws

A tuning law — the functional relationship between the Poincaré states,  $(\nu, \alpha)$ , and the clock period,  $T_c$  — effectively specifies a feedback controller. In this study we will consider a particular family of tuning laws,

$$T_c(k) := \tau(\nu_k, \alpha_k) := \tau_1(\nu_k) + \tau_2(\alpha_k), \quad (3.1)$$

which satisfy a fixed point condition,

$$\tau(\nu^*, \alpha^*) = T_n, \quad (3.2)$$

and a derivative condition that we will introduce in (3.7). In the above formula  $T_n$  is the “nominal clock period” and defines the fixed point speed,  $\nu^*$ , according to (3.5). A tuning procedure, which we will describe shortly, governs the adjustments to the shape configuration,  $\mathbf{o}$ , in conjunction with the tuning law,  $\tau(\nu, \alpha)$ .

### 3.1.2 A Tuning Procedure

Notice that variations of the clock period,  $T_c$ , alone elicit variations in the individual cell durations,  $\beta_i$ , according to (2.21). Those clock cells that partially and/or completely coincide with the stance mode,  $\Theta_i$ ,  $i = 1, 2, \dots, e, N$ , to which we refer as the “stance cells,” affect the behavior of the coupled system (note that the lift-off cell index,  $e$ , and the last cell index,  $N$ , are not necessarily consecutive). Consequently, the stance cell durations,  $\beta_i$ ,  $i = 1, 2, \dots, e, N$ , appear in the return map,  $\mathbf{r}(\mathbf{p})$ . On the other hand, those cells that occur during the aerial phase,  $\Theta_j$ ,  $i = e + 1, \dots, N - 1$ , the “aerial cells,” have no effect on the return map,  $\mathbf{r}(\mathbf{p})$ .

For the feedback implementation, we will adopt a specific *tuning procedure* that relates the period,  $T_c$ , and the shape configuration,  $\mathbf{o}$ . For computational reasons,

we prefer to keep the stance cell durations,  $\beta_i$ ,  $i = 1, 2, \dots, e, N$ , constant as the clock period,  $T_c$ , is adjusted by the tuning mechanism. Recall that the clock period,  $T_c$ , is the sum of the durations of all clock cells,  $\beta_i$ . Hence, to set a particular period some cell durations have to change. To realize this we will alter the durations of the aerial cells — those cells that do not affect the return map,  $\mathbf{r}(\mathbf{p})$ . To maintain the stance cell durations constant we will adjust the discontinuity points of the shape,  $\{\theta_i\}_{i=2}^N$ , in conjunction with the changes in the clock period,  $T_c$ .

$$\begin{aligned}
\theta_1(k) &:= 0, \\
\theta_{i+1}(k) &:= \frac{2\pi}{T_c(k)}\beta_i + \theta_i, & i = 2, 3, \dots, e \\
\theta_{i+1}(k) &:= (\theta_N - \theta_{e+1}) \frac{i-e}{N-2-e} + \theta_{e+1}, & i = e + 1, \dots, N - 2 \\
\theta_N(k) &:= 2\pi \left(1 - \frac{1}{T_c(k)}\beta_N\right) \\
\theta_{N+1}(k) &:= 2\pi.
\end{aligned}$$

Note that this tuning procedure is in fact an artifact of the particular parameterization of the clock controller, defined in Section 2.1.2. Since the above procedure renders the stance durations,  $\beta_i$ ,  $i = 1, 2, \dots, e, N$ , independent of the clock period,  $T_c$ , we will simply consider them as independent parameters of the return map,  $\mathbf{r}(\mathbf{p})$ .

## 3.2 The Closed-Loop Setting

### 3.2.1 Derivation of the Return Map

For our feedback discussions, we will consider a forward coupled system operating in a fundamental repeatable mode sequence,  $\sigma = \mathcal{GAG}(e, N)$ . We will refer to this setup as the “(underlying) open-loop system.” Let there be a shape configuration,  $\mathbf{o}$ , defining the durations,  $\beta_i$ , and the stiffness,  $\xi_i$ , for the stance cells,  $i = 1, 2, \dots, e, N$ , such that the return map of the open-loop system has a valid fixed point,  $\mathbf{p}^*$ . The

limit behavior of the open-loop system is defined by the period of the open-loop controller clock, which we will refer to as the “nominal clock period” and denote by  $T_n$ .

In our discussions, the goal of feedback will be to retain the fixed point,  $\mathbf{p}^*$ , at its original value but improve its stability properties. We will assume that the feedback controller is parameterized by the same shape configuration,  $\mathbf{o}$ , as the underlying open-loop setting.

The feedback controller alters the clock parameters,  $(T_c, \mathbf{o})$ , at the event of clock reset,  $\theta = 0$  — the Poincaré sampling event according to (2.18). Since the system parameters remain constant from one Poincaré sample to the next, the return map derivation procedure in Section 2.4.1 is immediately applicable to the closed-loop setup, as well. By virtue of the specific tuning procedure in Section 3.1, which keeps the stance cell durations constant,  $\beta_i = \text{const}$ ,  $i = 1, 2, \dots, e, N$ , the “closed-loop return map,”  $\mathbf{r}(\mathbf{p})$ , takes the following form

$$\mathbf{p}_{k+1} = \mathbf{r}(\mathbf{p}_k); \quad \mathbf{r}(\nu, \alpha) := \begin{bmatrix} \nu\rho(\alpha) \\ \tau(\nu, \alpha) - \beta_{[1, e-1]} - \varphi_e(\alpha) - 2\nu\rho(\alpha) \end{bmatrix}. \quad (3.3)$$

The primary feature of the closed-loop return map,  $\mathbf{r}(\mathbf{p})$ , is that the clock period,  $T_c$ , which was a constant in the open-loop return map, is now a function of the Poincaré states as described by the tuning law,  $\tau(\nu, \alpha)$ . It is important to note that the open-loop and closed-loop return maps have *identical* expressions for the forcing function,  $\rho(\alpha)$ , as well as the remaining time function,  $\varphi_e(\alpha)$ .

Since the hybrid structure of the continuous-time flow is preserved in the closed-loop setup, the arguments in Proposition 2 are applicable to the derivation of the valid domain,  $\mathcal{P}(\sigma)$ , of the closed-loop return map,  $\mathbf{r}(\mathbf{p})$ . Recall that both the valid delay interval,  $\mathcal{D}(\sigma)$ , and the invariant delay interval,  $\mathcal{D}^*(\sigma)$ , are governed by the

properties of the continuous-time flow of the coupled system during the stance mode,  $\mathcal{G}$ . As a direct result of the particular tuning procedure in Section 3.1 the stance flow of the closed-loop and open-loop settings are identical. Hence, the closed-loop setting and the underlying open-loop system share the same valid delay interval,  $\mathcal{D}(\sigma)$ , in (2.32) and the same invariant delay interval,  $\mathcal{D}^*(\sigma)$ , in (2.34). Consequently, for the closed-loop setup the valid Poincaré space,  $\mathcal{P}(\sigma)$ , is given by

$$\mathcal{P}(\sigma) := \{ (\alpha, \nu) \in \mathcal{P} \mid \alpha \in \mathcal{D}(\sigma) \wedge \pi_2 \circ \mathbf{r}(\mathbf{p}) \in (0, \min \{ \beta_N, \pi / \gamma_N \}) \}, \quad (3.4)$$

where  $\mathbf{r}(\mathbf{p})$ , is the closed-loop return map in (3.3). The discussions concerning specific tuning laws will derive explicit expressions for the valid Poincaré space,  $\mathcal{P}(\sigma)$ .

The fixed points of the closed-loop return map are given by

$$\begin{aligned} \rho(\alpha^*) &= 1, \\ \nu^* &= [\tau(\nu^*, \alpha^*) - \beta_{[1, e-1]} - \varphi_e(\alpha^*) - \alpha^*] / 2. \end{aligned} \quad (3.5)$$

Note that the result in Lemma 2 applies to the closed-loop fixed points as well. Since we assume the underlying open-loop system has a valid fixed point it follows that the closed-loop system will as well. Recall that the open-loop and closed-loop settings have identical forcing functions,  $\rho(\alpha)$ , and the invariant delay interval,  $\mathcal{D}^*(\sigma)$ . Furthermore, the open-loop and closed-loop fixed points are identical as a result of the fixed point condition in (3.2).

### 3.2.2 Local Stability Analysis

To assess the (local) stability properties of a given closed-loop fixed point,  $\mathbf{p}^*$ , we derive the Jacobian of the closed-loop return map,  $\mathbf{r}(\mathbf{p})$ , evaluated at this fixed point,

$$\mathbf{J}(\mathbf{p}^*) := D\mathbf{r}(\mathbf{p})|_{\mathbf{p}=\mathbf{p}^*} = \begin{bmatrix} 1 & \nu^* \rho'(\alpha^*) \\ \frac{d\tau_1(\nu^*)}{d\nu} - 2 & \frac{d\tau_2(\alpha^*)}{d\alpha} - \varphi_e'(\alpha^*) - 2\nu^* \rho'(\alpha^*) \end{bmatrix}. \quad (3.6)$$

which we will call the “closed-loop Jacobian” in (2.36). As a direct consequence of the similarities between the return maps of the open-loop and closed-loop settings the Jacobians of these two cases also resemble each other. We compute the closed-loop eigenvalues,  $\lambda_{\pm} := 1/2(\text{tr} \pm \sqrt{\Delta})$ , in terms of the closed-loop trace,  $\text{tr}$ , and the closed-loop discriminant,  $\Delta$ .

In order to take full advantage of the results from the open-loop analysis in Section 2.4.4, we will impose a set of derivative conditions,

$$\frac{d\tau_1(\nu^*)}{d\nu} = 0, \quad \frac{d\tau_2(\alpha^*)}{d\alpha} > \varphi_e'(\alpha^*). \quad (3.7)$$

which we will assume the tuning law,  $\tau(\nu, \alpha)$ , satisfies. Noting that the conditions in (3.7) imply that the partial derivative of the tuning law,  $\tau(\nu, \alpha)$ , with respect to the delay,  $\alpha$ , is independent of the speed state,  $\nu$ , we define the “modified remaining time function<sup>1</sup>,”

$$\tilde{\varphi}_e(\alpha) := \varphi_e(\alpha) - \tau_2(\alpha). \quad (3.8)$$

and compute the closed-loop trace,

$$\text{tr} := 1 - \tilde{\varphi}_e'(\alpha^*) - 2\nu^* \rho'(\alpha^*), \quad (3.9)$$

and the close-loop discriminant,

$$\Delta := [1 - \tilde{\varphi}_e'(\alpha^*) - 2\nu^* \rho'(\alpha^*)]^2 + 4\varphi_e'(\alpha^*), \quad (3.10)$$

---

<sup>1</sup>The modified remaining time function does not share the same physical interpretation as the original remaining time function.

which share the same analytic structure with their open-loop counterparts in (2.38) and (2.39), respectively. These two sets of expressions differ only in the function that governs their delay dependence. In the open-loop case, the delay dependence is captured by the first derivative of the remaining time function,  $\varphi_e'(\alpha)$ . In the closed-loop expressions the delay dependence is defined by the modified remaining time function,  $\tilde{\varphi}_e(\alpha)$ . Note also that the conditions in (3.7) guarantee that the modified remaining time function,  $\tilde{\varphi}_e(\alpha)$ , is a monotonically decreasing function of delay,  $\alpha$ , like the remaining time function,  $\varphi_e(\alpha)$ , in the open-loop case.

The strong similarities between the open-loop and closed-loop expressions combined with the strict monotonicity of the modified remaining time function,  $\tilde{\varphi}_e(\alpha)$ , lead to the conclusion that the closed loop trace,  $\text{tr}$ , and discriminant,  $\Delta$ , are related to the closed-loop trace bound,  $\overline{\text{tr}} := \text{tr}(T_n = T_c^0(\alpha^*))$ ,

$$\overline{\text{tr}} = 1 - \tilde{\varphi}_e'(\alpha^*),$$

and the closed-loop discriminant bound,  $\overline{\Delta} := \Delta(T_n = T_c^0(\alpha^*))$ ,

$$\begin{aligned} \overline{\Delta} &= (1 - \tilde{\varphi}_e'(\alpha^*))^2 + 4\tilde{\varphi}_e'(\alpha^*) \\ &= (1 + \tilde{\varphi}_e'(\alpha^*))^2, \end{aligned}$$

the same way the open-loop trace and discriminant are related to their respective bounds. Hence, the conclusions of the open-loop analysis, which are based on the relationships between the trace and the discriminant, are also valid for the closed-loop case. Instead of repeating those same derivations, we will simply summarize the stability properties of the closed-loop system with references to those comparable results in the open-loop analysis where the reader can find detailed derivations. As in Section 2.4.4 we will separately consider the three fixed point groups defined in (2.40).

For  $\mathbf{p}^* \in \mathcal{P}_-$  both the closed-loop trace,  $\text{tr}$ , and the closed loop discriminant,  $\Delta$ , are larger than their respective bounds,

$$\forall \mathbf{p}^* \in \mathcal{P}_- \quad \forall T_c \in \mathcal{T} \quad ( \Delta > \bar{\Delta} > 0 \quad \wedge \quad \text{tr} > \bar{\text{tr}} > 1 ),$$

and therefore,  $\lambda_+ = \frac{1}{2}(\text{tr} + \sqrt{\Delta}) > \frac{1}{2}(\bar{\text{tr}} + \sqrt{\bar{\Delta}}) = 1$  is located outside the unit circle rendering all fixed points  $\mathbf{p}^*$ , in  $\mathcal{P}_-$  unstable. In the closed-loop setup the analytic structure of the forcing function,  $\rho(\alpha)$ , and its role in the return map,  $\mathbf{r}(\mathbf{p})$ , as a scaling factor for the energy map are preserved. Therefore, the arguments in Section 2.4.4 concerning the stability properties of  $\mathbf{p}^* \in \mathcal{P}_0^*$  are applicable to the closed-loop setting as well. Hence, we conclude that all fixed points,  $\mathbf{p}^*$ , in  $\mathcal{P}_0^*$  are also unstable.

Next, we will consider fixed points,  $\mathbf{p}^*$ , in  $\mathcal{P}_+^*$  and compute the locus of the closed-loop eigenvalues,  $\lambda_{\pm}$ , as a function of the nominal clock period,  $T_n$ , which is related to the fixed point speed,  $\nu^*$ , by the affine relationship in (3.5) and the tuning law condition in (3.2). The locus of the closed-loop eigenvalues,  $\lambda_{\pm}$ , is parameterized by the first derivative of the modified remaining time function evaluated at the fixed point delay,  $\varphi_e'(\alpha^*)$ . See Figure 3.2 for an illustration of the eigenvalue locus in the closed-loop settings.

At the smallest admissible nominal period,  $T_n = T_c^0$ , the two eigenvalues are real and located at  $\lambda_-(T_c^0) = -\tilde{\varphi}'_e(\alpha^*)$  and  $\lambda_+(T_c^0) = 1$ . As the nominal clock period,  $T_n$ , increases, the two eigenvalues converge towards each other along the real axis and meet at  $\lambda_{\pm}(T_c^1) = \sqrt{-\tilde{\varphi}'_e(\alpha^*)}$ . For nominal clock periods  $T_n \in \mathcal{T}_c$ , the two eigenvalues are complex conjugate,  $\lambda_{\pm} \in \mathbb{C}$ , and their magnitudes are specified by the first derivative of the modified remaining time function evaluated at the fixed point delay,

$$|\lambda_{\pm}| = \frac{1}{2} \sqrt{\text{tr}^2 - \Delta} = \sqrt{-\tilde{\varphi}'_e(\alpha^*)} \quad (3.11)$$

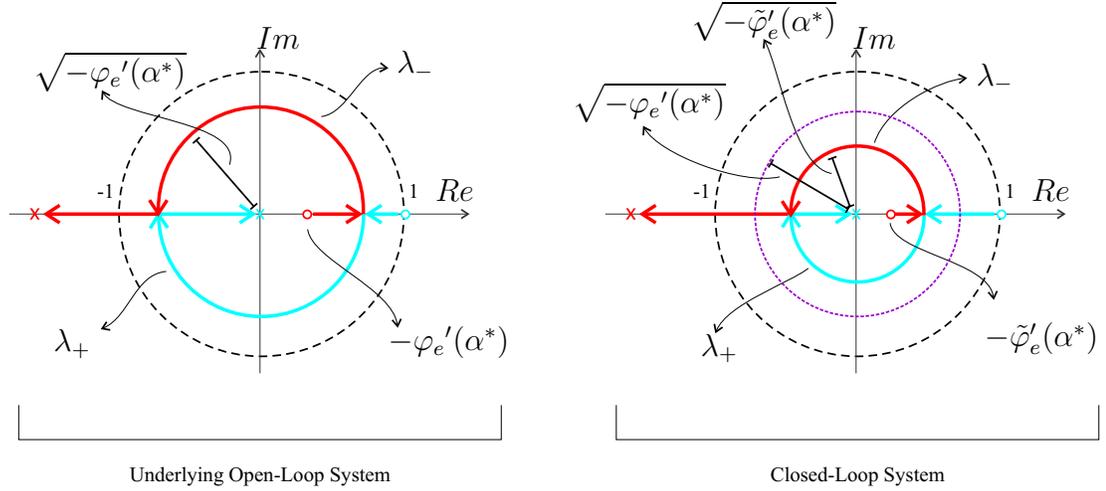


Figure 3.2: A conceptual comparison between the locus of the eigenvalues of open-loop (left) and synchronizer closed-loop (right) systems for a fixed point,  $\mathbf{p}^*$ , in  $\mathcal{P}_+^*$  class. A tuning law,  $\tau(\nu, \alpha)$ , which satisfies the conditions in Proposition 6 results in closed-loop eigenvalues that are closer to the origin than the eigenvalues of the underlying open-loop system, as depicted in the sketch on the right.

As the nominal period,  $T_n$ , continues to increase the two eigenvalues,  $\lambda_+$  and  $\lambda_-$ , move towards the left hand plane following the upper and lower half circular paths, respectively. At  $\lambda_{\pm}(T_c^2) = -\sqrt{-\tilde{\varphi}'_e(\alpha^*)}$  the eigenvalues become real again. As the nominal period,  $T_n$ , increases  $\lambda_+$  moves along the real axis towards the origin. On the other hand,  $\lambda_-$  converges to  $-\infty$ . Assuming that  $\tilde{\varphi}'_e(\alpha^*) < 1$ ,  $\lambda_-$  leaves the unit circle at  $T_c^3(\alpha^*) := \lambda_-^{-1}(-1)$ .

## 3.3 Improving the Rate of Convergence

### 3.3.1 Motivation

In the fundamental repeatable mode sequences,  $\mathcal{GAG}(e, N)$ , the delay state,  $\alpha$ , represents the *phase difference* between the controller clock and the “mechanical clock,” the hopper, sampled at the clock reset,  $\theta = 0$ . At a fixed point,  $\mathbf{p}^*$ , the delay state,  $\alpha_k = \alpha^*$  remains constant after each iteration of the return map,  $\mathbf{r}(\mathbf{p})$ . In other words, the phase difference between the controller clock and the mechanical clock remains constant at a particular event indicating that these two cyclic systems are *synchronized*.

The forward coupling in Section 2.1.3 offers a mechanism to alter the total mechanical energy,  $H$ , as a function of the delay state,  $\alpha$ . In the return map,  $\mathbf{r}(\mathbf{p})$ , this feature is captured by the forcing function,  $\rho(\alpha)$ . It follows from the properties of the hybrid potential energy of the hopper in (2.1.1) (a combination of the Hook’s law spring law and the gravitational potential) that the period of hopping, the mechanical clock period,  $T_m$ , is a monotonically increasing function of the total mechanical energy,  $H$ . Hence, the controller actions over a cycle (modulation of the spring stiffness) not only change the total mechanical energy of the hopper,  $H$ , but also change the period of the mechanical clock,  $T_m$ .

In the forward coupled system, synchronization of the mechanical clock and the controller clock is achieved by pulling the period of the mechanical clock,  $T_m$ , toward the constant period of the controller clock,  $T_c$ , by changing the mechanical energy,  $H$ . However, the authority of the controller over the mechanical clock period,  $T_m$ , is indirect, as explained above, and therefore, synchronization can only occur slowly. This feature of the forward coupled system was demonstrated in the numerical results in Section 2.5.3.

On the other hand, the controller clock in Section 2.1.2, whose parameters can be changed at will, is completely under the user’s control. We observe that the synchronization of the controller clock and mechanical clock can be achieved much more effectively, and in fewer cycles, if the period of the controller clock,  $T_c$ , is also altered in as a function of the delay state,  $\alpha$ . That is, the controller clock can assist the synchronization instead of remaining inactive.

### 3.3.2 A Family of Synchronizer Tuning Laws

We posit a family of tuning controllers,

$$\tau(\nu, \alpha) := T_n + \tau_{sync}(\alpha - \alpha^*) \quad (3.12)$$

where the “synchronization function,”  $\tau_{sync}(s)$ , satisfies

$$\tau_{sync}(0) = 0, \quad \tau'_{sync}(0) > \varphi_e(\alpha^*). \quad (3.13)$$

We will refer to the members of the family of tuning laws in (3.12) as “synchronizers.” These discrete-time feedback controllers adjust the clock period in the  $k^{\text{th}}$  cycle,  $T_c(k)$ , as a function of the phase error which we represent by the difference between the  $k^{\text{th}}$  cycle delay,  $\alpha_k$  and the fixed point delay,  $\alpha^*$ . By direct computation, it can be shown that synchronizers satisfy the tuning law conditions in (3.2) and (3.7). Hence, the analysis in Section 3.2.2 is applicable to the synchronizers.

Since the tuning law in (3.12) is a function of the delay state,  $\alpha$ , only, we can derive analytic boundaries for the closed-loop valid domain,  $\mathcal{P}(\sigma)$ , by properly rearranging the condition on the second entry of the closed-loop return map in (3.4) which leads to bounds on the speed state,  $\nu$ , that are parameterized in terms of the delay state,  $\alpha$ ,

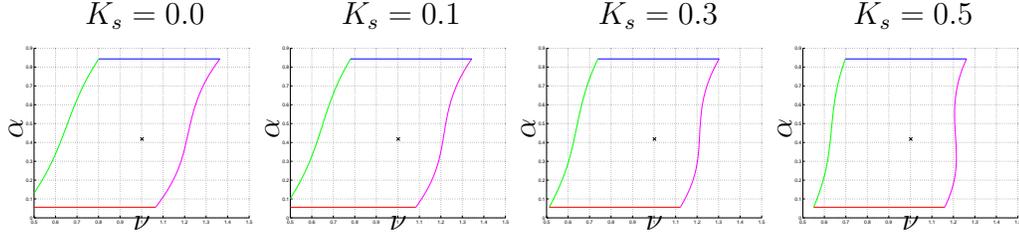


Figure 3.3: The valid domain,  $\mathcal{P}(\sigma)$ , of the closed-loop system employing the linear synchronizer tuning law in (3.15) at different synchronization gains,  $K_s$ . Note that the valid delay interval,  $\mathcal{D}(\sigma)$ , remains unchanged, whereas the bounds on the speed state,  $\nu$ , gets shifted as a result of the changing synchronization gain,  $K_s$

$$V(\alpha, 0) > \nu > V(\alpha, \min\{\beta_N, \pi/\gamma_N\})$$

where

$$V(\alpha, s) := \frac{T_n + \tau_{sync}(\alpha - \alpha^*) - \beta_{[1, e-1]} - \tilde{\varphi}_e(\alpha) - s}{2\rho(\alpha)}. \quad (3.14)$$

Figure 3.3 illustrates the valid domain of a closed-loop system that employs a linear synchronizer defined in (3.15), below. Notice that the synchronizer has no effect on the valid delay interval,  $\mathcal{D}(\sigma)$ . However, the bounds on the speed state,  $\nu$ , are skewed by the introduction of the tuning law. This relationship gives the designer some degree of control over the shape of the valid domain,  $\mathcal{P}(\sigma)$ , which will prove to be useful in the sequential composition of valid domains in Section 3.4.

**Proposition 6.** *Consider a fixed point,  $\mathbf{p}^* \in \mathcal{P}_+^*$ , and a nominal clock period,  $T_n \in \mathcal{T}_c$ , where the eigenvalues of this fixed point are complex,  $\lambda_{\pm} \in \mathbb{C}$ . The magnitude of the closed loop eigenvalues,  $|\lambda_{\pm}|$ , monotonically decreases as a function of the derivative of the synchronization function,  $\tau'_{sync}(\alpha)$ , evaluated at the fixed point delay,  $\alpha^*$ .*

PROOF. The result follows from the results in Section 3.2.2 where we derived the locus of the closed-loop eigenvalues for fixed points,  $\mathbf{p}^*$ , in the  $\mathcal{P}_+^*$  group. For  $T_n \in \mathcal{T}_c$ , the eigenvalues are complex,  $\lambda_{\pm}$ , and located on a circle centered at the origin whose radius in (3.11) can be expanded using the definitions in (3.8) and (3.12),

$$|\lambda_{\pm}| = \sqrt{-\tilde{\varphi}'_e(\alpha^*)} = \sqrt{-\tilde{\varphi}_e(\alpha^*) + \tau'_{sync}(\alpha^*)}$$

which decreases monotonically with the derivative of the synchronization function evaluated at the fixed point delay,  $\tau'_{sync}(\alpha^*)$ . □

Since eigenvalues with smaller real part imply smaller time constants for the convergence to the fixed point it follows from Proposition 6 that a properly designed synchronizer tuning law in (3.12) improves the rate of convergence locally. Typically, improvement of the local stability extends to the behavior of the coupled system at larger scales. We will demonstrate this in a series of numerical studies shortly. Another immediate corollary to Proposition 6 is that the synchronizer tuning law in (3.12) renders the lossless settings asymptotically stable. Recall that the lossless cases were only neutrally stable in the open-loop case.

### 3.3.3 A Numerical Demonstration: Linear Synchronizer

We will demonstrate the application of the synchronizer tuning law in (3.12) in a series of numerical studies where we will consider a linear synchronization function,

$$\tau_{sync}(\alpha) := -K_s \alpha \tag{3.15}$$

where  $K_s \in \mathbb{R}$  is the “synchronization gain” which is bounded by the first derivative of the remaining time function evaluated at the fixed point,

$$K_s < -\varphi_e'(\alpha^*), \quad (3.16)$$

in accordance with (3.13).

The magnitude of the synchronization gain,  $K_s$ , effectively determines the strength of the active synchronization that the tuning mechanism will perform. Furthermore, its magnitude is also determines the convergence properties of the closed-loop system according to Proposition 6. Consider those nominal clock periods,  $T_n \in \mathcal{T}_c$ , that give rise to complex eigenvalues,  $\lambda_{\pm} \in \mathbb{C}$ , whose magnitude can be computed in terms of the synchronization gain,  $K_s$ , and the first derivative of the remaining time function evaluated at the fixed point delay,  $\varphi_e'(\alpha^*)$ ,

$$|\lambda_{\pm}| = \sqrt{-\varphi_e'(\alpha^*) - K_s} \quad (3.17)$$

Zero gain,  $K_s = 0$ , corresponds to the open-loop case. It directly follows that for positive gain values,  $K_s > 0$ , the eigenvalues move closer to the origin, effectively decreasing the time constant of convergence in the closed-loop system. Figure 3.4 demonstrates the change in the convergence speed in the closed-loop system as a function of the synchronization gain,  $K_s$ . As the synchronization gain,  $K_s$ , gets close to its upper bound in (3.16) the eigenvalues converge to the origin and the control takes an dead-bead nature (see Figure 3.4 bottom-left ). Naturally, negative values of the synchronization gain,  $K_s < 0$ , have detrimental effects on the closed-loop system. We ignore the negative cases.

We do not have any closed form expression for the basin of the coupled system. Instead, we numerically evaluated the basin for the closed-loop system that uses the tuning law in (3.15). Figure 3.5 exhibits the basin of the closed-loop system at various settings of the synchronization gain,  $K_s$ . Note that the tuning law in (3.15) not only improves the convergence speed but also enlarges the domain of attraction.

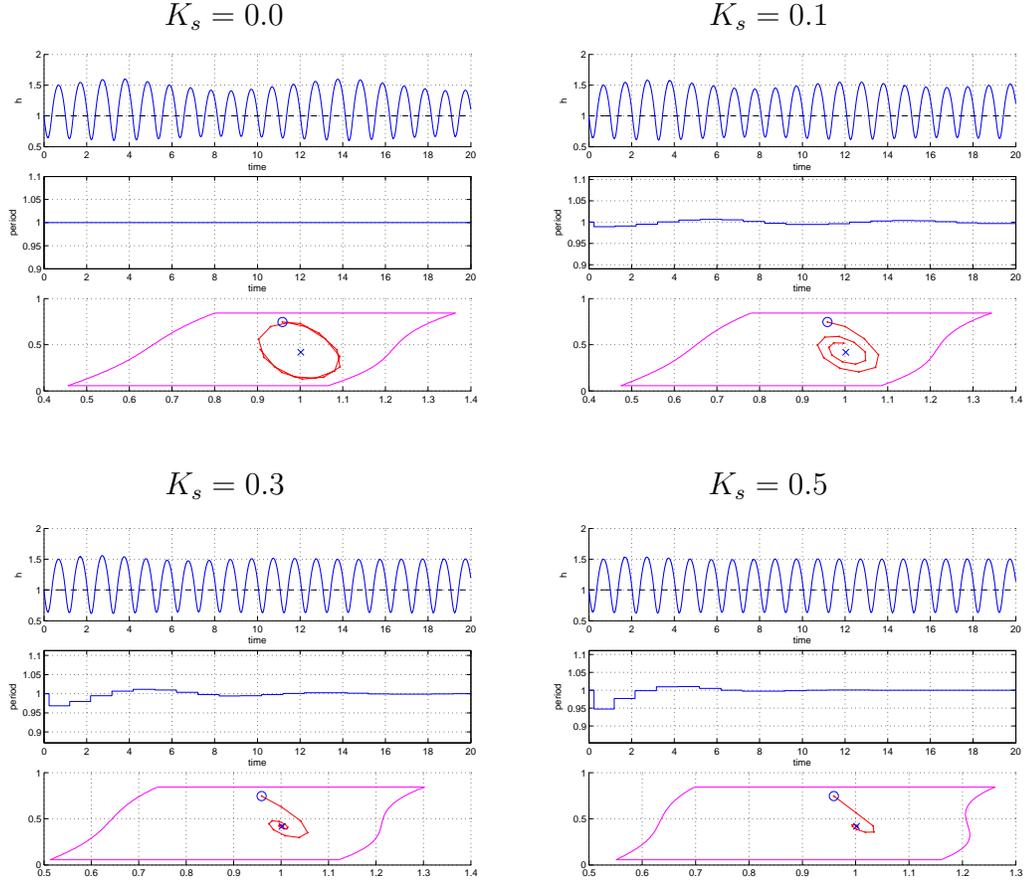


Figure 3.4: Convergence in a closed-loop system that uses the tuning law in (3.15). The effect of different synchronization gains,  $K_s$ , is demonstrated. Each subplot contains: body height (top); instantaneous clock period,  $T_c$  (middle); and the Poincaré sequence,  $\mathbf{p}_k$  (bottom). The bounded area in the Poincaré space is the valid domain,  $\mathcal{P}(\sigma)$ , of the closed-loop return map,  $\mathbf{r}(\mathbf{p})$ .

## 3.4 Enlarging the Basin of Attraction

### 3.4.1 Motivation

One major shortcoming of the open-loop setting is that the valid domain,  $\mathcal{P}(\sigma)$ , of a typical fundamental repeatable mode sequence,  $\sigma = \mathcal{G}\mathcal{A}\mathcal{G}(e, N)$ , is small. By

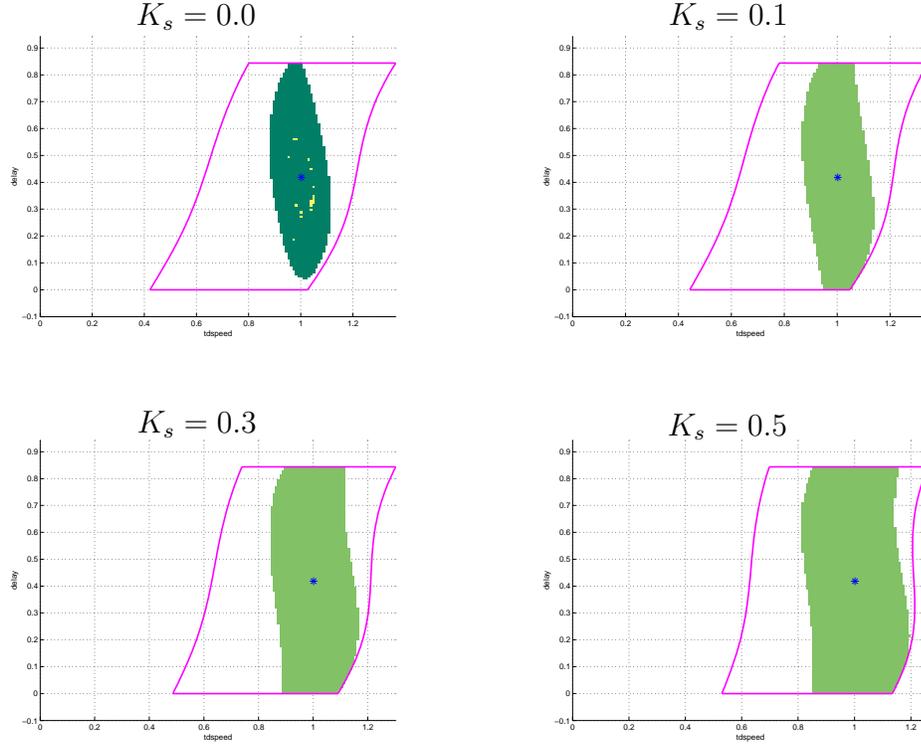


Figure 3.5: The domain of attraction of a closed-loop system that uses the tuning law in (3.15). The subplots exhibit the basins (solid filled) and the valid domain (bounded area) at different settings of the synchronization gain,  $K_s$ .

definition, the basin of a fixed point,  $\mathbf{p}^*$ , is contained within the valid domain,  $\mathcal{P}$ . Therefore, the limitations on the valid domain,  $\mathcal{P}(\sigma)$ , directly constrain the basin of the stable fixed points,  $\mathbf{p}^*$ , and ultimately results in a system that is very sensitive to external disturbances. Naturally, such sensitivity is not desirable in practical applications.

From an intuitive point of view, one can argue that the validity of the mode sequence can be maintained if the clock period,  $T_c$ , is continuously matched to that of the mechanical hopping,  $T_m$ . Recall that the hopping period,  $T_m$ , in this 1-DOF mechanism is directly related to its energy, which is effectively represented by the

speed state,  $\nu$ , in the Poincaré space. This observation is the basis of the tuning mechanism that we will introduce in this section.

Indeed, the analysis of the open-loop return map concludes that for a fundamental repeatable mode sequences,  $\sigma = \mathcal{GAG}(e, N)$ , the positions of both the valid domain,  $\mathcal{P}(\sigma)$ , (defined in Section 2.31) and the fixed point,  $\mathbf{p}^*$ , (defined in (2.33)) are parameterized by the clock period,  $T_c$ . Hence, by varying the clock period,  $T_c$ , both the valid domain,  $\mathcal{P}$ , and the fixed point,  $\mathbf{p}^*$ , that it contains can be translated along the speed axis,  $\nu$ , without affecting the existence and the validity of the fixed point,  $\mathbf{p}^*$ . In fact, the fixed point delay,  $\alpha^*$ , remains unchanged as the clock period,  $T_c$ , varies. Naturally, the basin of attraction, which is located in the neighborhood of the fixed point,  $\mathbf{p}^*$ , is also translated along with the fixed point,  $\mathbf{p}^*$ , when the clock period,  $T_c$ , is changed.

This feature of the open-loop return map inspires a sequential composition controller to enlarge the effective basin of a given fixed point,  $\mathbf{p}^*$ . The basic idea is to sense the speed state at the beginning of a clock cycle,  $\nu_k$ , and compute the clock period for that cycle,  $T_c(k)$ , such that the  $k^{\text{th}}$  cycle fixed point speed,  $\nu_k^*$ , is located between the target fixed point speed,  $\nu^*$ , and the current speed,  $\nu_k$ . Naturally, the  $k^{\text{th}}$  cycle fixed point,  $\mathbf{p}_k^*$ , needs to be placed close enough to the current state,  $\mathbf{p}_k$ , so that the  $k^{\text{th}}$  cycle basin captures the current state,  $\mathbf{p}_k$ . The constant slight offset towards the target fixed point,  $\mathbf{p}^*$ , effectively pulls the states towards this desired operating point,  $\mathbf{p}^*$ .

### 3.4.2 A Sequential Composition Procedure

We consider the open-loop return map,  $\mathbf{r}(\mathbf{p}, T_c)$ , as a one-parameter family of maps parameterized by the clock period,  $T_c$ . Naturally, the valid domain,  $\mathcal{P}(\sigma, T_c)$ , the fixed point(s),  $\mathbf{p}^*(T_c)$ , and the basin of the stable fixed point,  $\mathcal{P}^*(\sigma, T_c)$ , are all

parameterized by the clock period,  $T_c$ , as well. Figure 3.6 illustrates the displacement of the valid domain,  $\mathcal{P}(\sigma, T_c)$ , and the fixed point,  $\mathbf{p}^*(T_c)$ , as a function of the clock period,  $T_c$ . We will define a composition procedure that will “properly” pick a member of this return map family for every clock cycle such that the validity of the cycle Poincaré state,  $\mathbf{p}_k$ , is satisfied and the Poincaré sequence is steered towards a desired operating regime that is described by the “target fixed point,”  $\mathbf{p}^* = (\nu^*, \alpha^*)^T$ .

Figure 3.6: A conceptual illustration of the dependence between the clock period,  $T_c$ , and the location of the valid domain,  $\mathcal{P}(\sigma, T_c)$ , and the fixed point,  $\mathbf{p}^*(T_c)$ .

First, we will introduce a set of naming conventions to aid our explanations. The  $k^{\text{th}}$  cycle refers to the time interval between the  $k^{\text{th}}$  and  $(k+1)^{\text{th}}$  clock reset events. The composition tuning law, which we will refer in short as the “composer,” sets the “ $k^{\text{th}}$  cycle clock period,”  $T_c(k)$ , as a function of the  $k^{\text{th}}$  cycle speed state,  $\nu_k$ , that is sampled at the  $k^{\text{th}}$  clock reset. This effectively selects a member of the return map. We define the “ $k^{\text{th}}$  cycle valid domain,”  $\mathcal{P}_k(\sigma)$ , as the subset of the Poincaré space,  $\mathcal{P}$ , characterized by (3.4) evaluated for the  $k^{\text{th}}$  cycle clock period,  $T_c(k)$ . Similarly, the solution to the fixed point equation in (3.5) for the  $k^{\text{th}}$  cycle period,  $T_c(k)$ , will be referred as the “ $k^{\text{th}}$  cycle fixed point” and will be denoted by  $\mathbf{p}_k^* = (\nu_k^*, \alpha_k^*)^T$ . The basin of the  $k^{\text{th}}$  cycle fixed point will be denoted by  $\mathcal{P}_k^*(\sigma)$ . Figure 3.7 graphically illustrates these conventions.

To implement the composition we posit a tuning controller,  $\tau(\nu, \alpha) = \tau_{comp}(\nu)$ ,

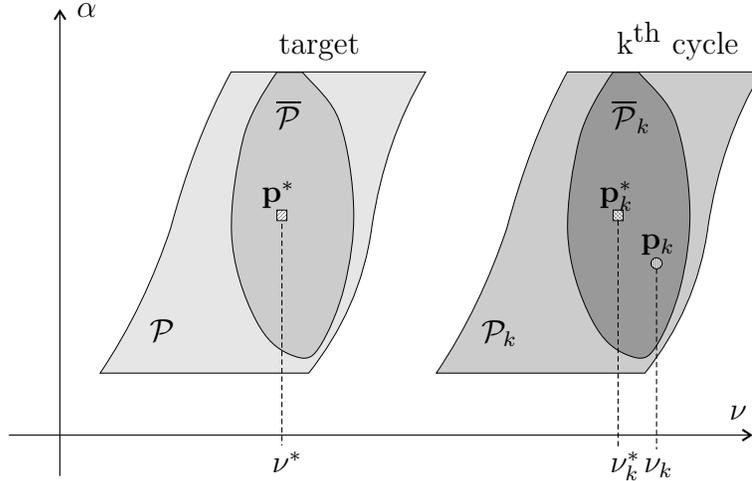


Figure 3.7: An illustration of the naming convention employed in the composition discussion. Note how the changes in the clock period lead to translations in the valid domain,  $\mathcal{P}_k(\sigma)$ , fixed point,  $\mathbf{p}_k^*$ , and the basin,  $\mathcal{P}_k^*$ . The target related expressions are identified by the lack of subscripts.

$$\tau_{comp}(\nu) := \begin{cases} 2 \left[ \nu - K_c \arctan \left( \frac{\nu - \nu^*}{K_c} \right) \right] + \beta_{[1, e-1]} + \varphi_e(\alpha^*) + \alpha^* & ; K_c \neq 0 \\ T_n & ; K_c = 0 \end{cases} \quad (3.18)$$

where  $K_c$  is a scalar “composition gain.” The tuning law in (3.18) satisfies the tuning law conditions in (3.2) and (3.7) which can be shown by direct computation. Hence, the results of Section 3.2.2 applies the the closed-loop system that employs the tuning law in (3.18).

It is important to note that the composition tuning law in (3.18) is independent of the delay state,  $\alpha$ , hence,  $\tilde{\varphi}'_e(\alpha) \equiv \varphi'_e(\alpha)$ . This implies that the stability of the target fixed point,  $\mathbf{p}^*$ , is governed by the underlying system which we will assume to be stable.

Assuming that the underlying system<sup>2</sup> is the open-loop setting described in Section 2.1.3 the  $k^{\text{th}}$  cycle fixed point,  $\mathbf{p}_k^*$ , can be evaluated by substituting the composition tuning law in (3.18) into the fixed point formula in (2.33) which yields

$$\mathbf{p}_k^* = \begin{bmatrix} \nu_k - K_c \arctan\left(\frac{\nu_k - \nu^*}{K_c}\right) \\ \alpha^* \end{bmatrix} \quad (3.19)$$

The positive values of the composition gain,  $K_c > 0$ , place the  $k^{\text{th}}$  cycle fixed point,  $\mathbf{p}_k^*$  between the target fixed point,  $\mathbf{p}^*$ , and current state,  $\mathbf{p}_k$ . For negative values of the composition gain,  $K_c < 0$ , the  $k^{\text{th}}$  cycle fixed point,  $\mathbf{p}_k^*$ , the target fixed point,  $\mathbf{p}^*$ , and the  $k^{\text{th}}$  cycle fixed point,  $\mathbf{p}_k^*$ , are located at the opposite sides of the current state,  $\mathbf{p}_k$ , which causes the Poincaré sequence,  $\{\mathbf{p}_k\}_{k=0}^{\infty}$ , to diverge from the target fixed point,  $\mathbf{p}^*$ . Hence, we will only concentrate on non-negative composition gain,  $K_c \geq 0$ . Setting  $K_c = 0$  disables the composer. The distance between the  $k^{\text{th}}$  cycle fixed point,  $\mathbf{p}_k^*$ , and  $k^{\text{th}}$  cycle state,  $\mathbf{p}_k$ , monotonically increases as a function of the composition gain,  $K_c$ . The reader can refer to Figure 3.7 for a graphical illustration.

Proposition 7 will show that under certain conditions the composition tuning law in (3.18) expands the effective valid domain,  $\mathcal{P}$ , of the closed loop system to span the entire speed axis,  $\nu$ . The immediate physical significance is that the closed-loop system can address all period-one hopping gaits.

**Proposition 7.** *Let  $\mathbf{r}(\mathbf{p}, T_c)$ , be the one-parameter family of return maps of an underlying system, and  $\mathbf{p}^*$ , be a valid stable fixed point of this return map. If the return map satisfies*

---

<sup>2</sup>We will consider a case in Section 3.5 where the underlying system is another closed-loop system that offers the same period parameterization as the open-loop case.

$$\begin{aligned}
\forall \mathbf{p} \in \{\mathcal{P} \mid \alpha \in \mathcal{D}(\sigma) \wedge \pi_2 \circ \mathbf{r}(\mathbf{p}) = \max\{0, \varphi_e^{-1}(\beta_e)\}\} & \quad \nu > \nu^* \\
\forall \mathbf{p} \in \{\mathcal{P} \mid \alpha \in \mathcal{D}(\sigma) \wedge \pi_2 \circ \mathbf{r}(\mathbf{p}) = 0\} & \quad \nu < \nu^*
\end{aligned} \tag{3.20}$$

for all admissible clock periods,  $T_c \in \mathcal{T}(\sigma) = (T_c^0, \infty)$ , then there exists a sufficiently small composition gain,  $K_c > 0$ , that gives rise to an effective closed-loop valid domain,

$$\mathcal{P}(\sigma) = \{\mathcal{P} \mid \alpha \in \mathcal{D}(\sigma)\}, \tag{3.21}$$

that spans the entire speed axis,  $\nu$ , that is for any  $\nu$ , there exists a delay,  $\alpha$ , such that  $(\nu, \alpha) \in \mathcal{P}(\sigma)$ .

PROOF. Consider the  $k^{\text{th}}$  cycle. In essence, the condition in (3.20) describes a situation where there exists an interval of speed,  $(\nu_k^{\min}, \nu_k^{\max})$ , that contains the  $k^{\text{th}}$  fixed point speed,  $\nu_k^*$ , such that the cross product between this speed interval,  $(\nu_k^{\min}, \nu_k^{\max})$ , and the valid delay interval,  $\mathcal{D}(\sigma)$ , is a subset of the valid domain,  $\mathcal{P}_k(\sigma)$ . We will refer to this rectangular area as a “strip.” Figure 3.8 illustrates this situation.

The composition tuning law in (3.18) places the  $k^{\text{th}}$  cycle fixed point,  $\mathbf{p}_k^*$ , in the vicinity of the  $k^{\text{th}}$  cycle Poincaré state,  $\mathbf{p}_k$ , as described by (3.19). For a constant composition gain,  $K_c$ , the difference between the  $k^{\text{th}}$  cycle fixed point speed,  $\nu_k^*$ , and  $k^{\text{th}}$  cycle Poincaré state speed,  $\nu_k$ , is bounded,  $\max[|\nu_k^* - \nu_k|] = K_c\pi/2$ .

Hence, by setting the composition gain,  $K_c$ , to a sufficiently small value it can be guaranteed that the  $k^{\text{th}}$  Poincaré state,  $\mathbf{p}_k$ , falls within  $(\nu_k^{\min}, \nu_k^{\max}) \times \mathcal{D}(\sigma)$ , which is a subset of the  $k^{\text{th}}$  valid space  $\mathcal{P}_k(\sigma)$ . Since the composition tuning law will enforce this condition at all cycles the effective valid space of the closed-loop system is the union of all possible strips which yields (3.21).

□

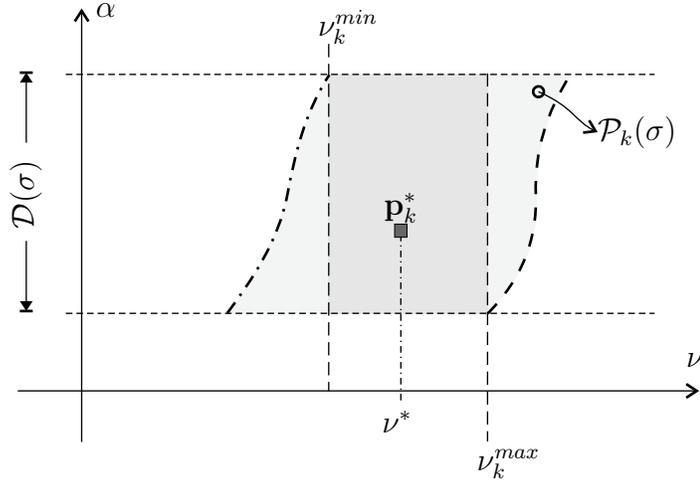


Figure 3.8: An illustration of the relative positions of the boundaries of the  $k^{\text{th}}$  valid domain,  $\mathcal{P}_k(\sigma)$ , the  $k^{\text{th}}$  fixed point,  $\mathbf{p}_k^*$ , and the  $k^{\text{th}}$  Poincaré state,  $\mathbf{p}_k$ . This sketch shows a case where the target fixed point speed,  $\nu^*$ , is smaller than the  $k^{\text{th}}$  cycle speed,  $\nu$ .

The conditions in (3.20) may not be satisfied by a given open-loop system. Although the shape configuration,  $\mathbf{o}$ , has a certain authority over the boundaries of the valid domain,  $\mathcal{P}(\sigma)$ , it is hard to determine a shape configuration,  $\mathbf{o}$ , that offers the desired stability conditions and a desired valid domain,  $\mathcal{P}(\sigma)$ . The discussion on shape design of Appendix B presents some of these difficulties. Instead, one can utilize the synchronization tuning law in Section 3.3.2 to satisfy the conditions in (3.20). Recall that the synchronization function,  $\tau_{sync}(\alpha)$ , has an immediate effect on the valid space form through (3.14). In Section 3.5 we will discuss how to use the synchronization and composition tuning laws in conjunction.

The composition tuning law in (3.18) not only stretches the effective valid domain of the closed-loop system to span the entire speed axis (for properly small composition gains) but also considerably increases the size of the basin of attraction. Figure 3.9

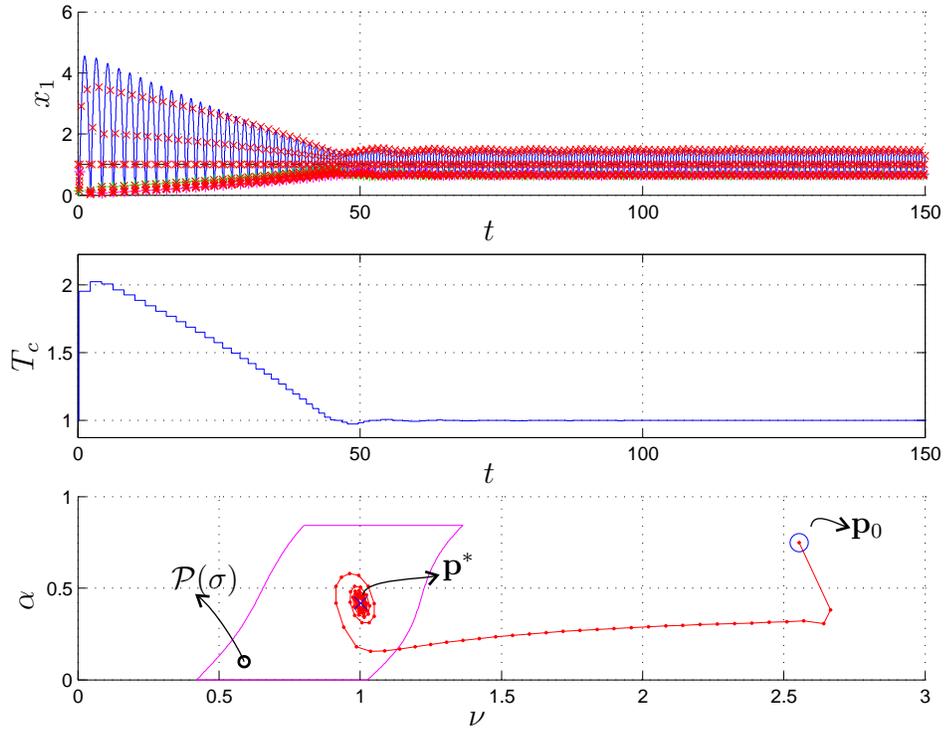


Figure 3.9: A typical run of the close-loop system employing the composition tuning law in (3.18). The figure contains three plots: the body height of the mechanical hopper,  $x_1$ , (top); the instantaneous clock period,  $T_c$ , as it is dictated by the composition tuning law (middle); and the the discrete flow of the Poincaré states,  $\mathbf{p}_k$ . The bottom plot indicates the valid domain of the open-loop system,  $\mathcal{P}(\sigma)$ .

depicts a typical run of the closed-loop system that employs the composition tuning law. Note that the initial condition,  $\mathbf{p}_0$ , is far outside the valid domain of the original open-loop system,  $\mathcal{P}(\sigma)$ . Yet, the cycle to cycle adjustments to the clock period,  $T_c$ , properly locate the cycle valid domains to capture and steer the Poincaré flow towards the desired operating regime,  $\mathbf{p}^*$ . Figure 3.10 provides a list of snapshots of along the run illustrated in Figure 3.9. The cycle valid domains,  $\mathcal{P}_k(\sigma)$ , and how it

is related to the cycle states,  $\mathbf{p}_k$ , is pointed out.

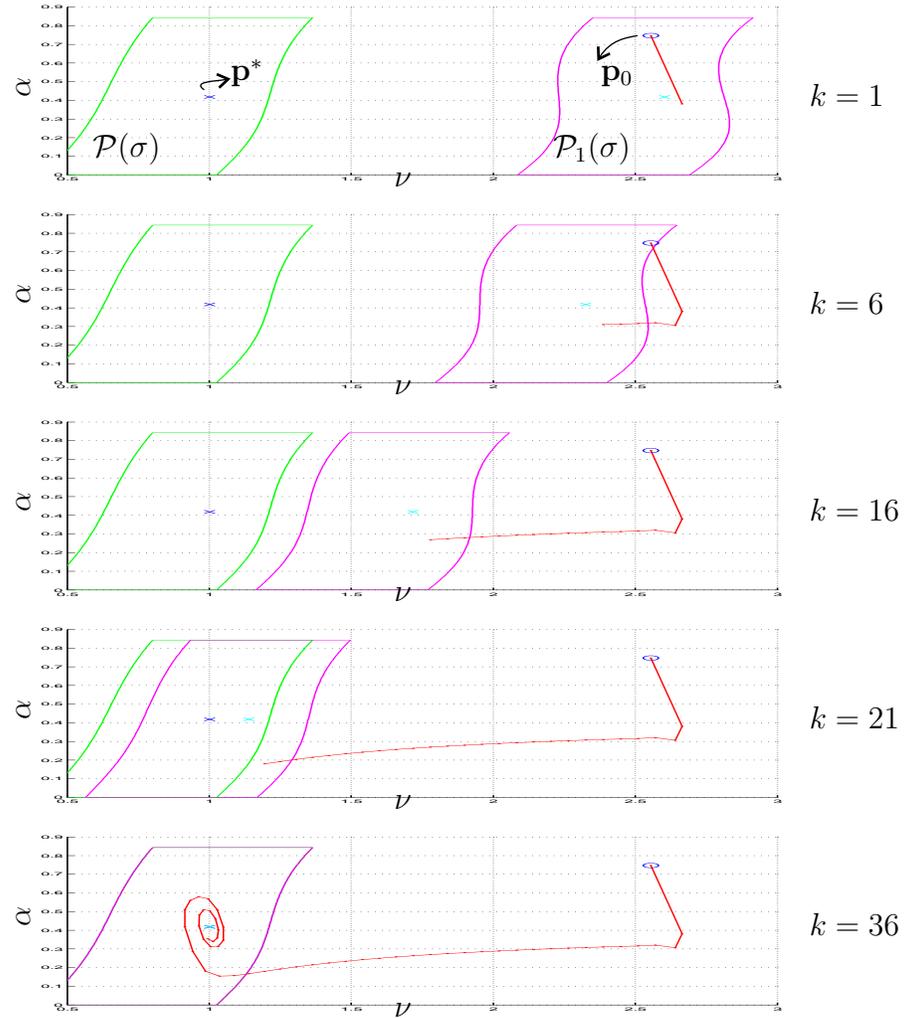


Figure 3.10: A list of snapshots during a typical run of a closed-loop system that employs the composition tuning law in (3.18). The far left column indicates the cycle index. Each plot represents the state of the closed-loop system at a particular cycle and contains the valid domain of the open-loop system,  $\mathcal{P}(\sigma)$ ; target fixed point,  $\mathbf{p}^*$ ; cycle valid domain,  $\mathcal{P}_k(\alpha)$ ; the cycle fixed point,  $\mathbf{p}_k^*$ ; and the cycle state,  $\mathbf{p}_k$ .

## 3.5 A Hybrid Tuning Law

The closed-loop system described in Section 3.3 is parameterized by the nominal clock period,  $T_n$ , the same way the open-loop system is parameterized by the clock period,  $T_c$ . Namely, the nominal clock period,  $T_n$ , dictates the location of the closed-loop valid domain and the fixed points. This feature allows us to augment the closed-loop system in Section 3.3 with the composition controller that we defined in Section 3.4 forming a hierarchical feedback controller as depicted in Figure 3.11. The tuning law in the hybrid controller takes the following form

$$\tau(\nu, \alpha) = \tau_{comp}(\nu) + \tau_{sync}(\alpha - \alpha^*), \quad (3.22)$$

which satisfies the conditions in (3.2) and (3.7). A graphical illustration of the hierarchical structure is presented in Figure 3.11

In this numerical demonstration, we will consider a hybrid tuning law consisting of the linear synchronizer in (3.15) and the composition manager in (3.18). The improvement in the convergence provided by the synchronizer allows us to increase composition gain which in turn increases the overall convergence speed. Figure 3.12 compares two cases with different gain settings to demonstrate the improvement in the overall behavior.

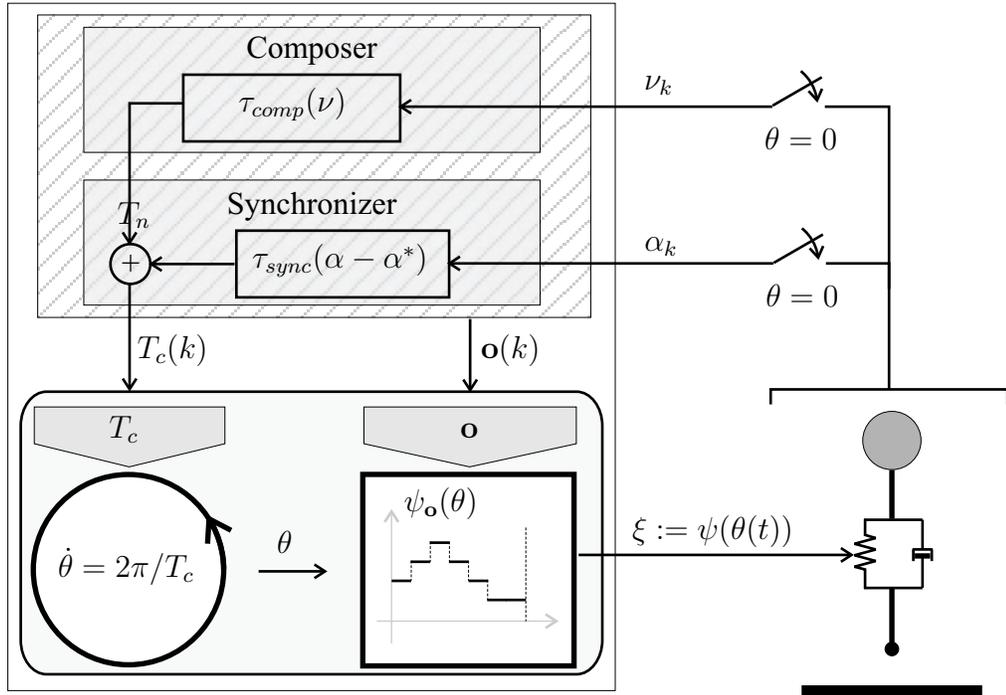


Figure 3.11: The hierarchical structure of the hybrid tuning law in (3.22). The synchronizer input, the nominal period,  $T_n$ , replaces the the original period input of the clock controller and offers the same parameterization of the valid domain,  $\mathcal{P}(\sigma)$ , and fixed point,  $\mathbf{p}^*$ .

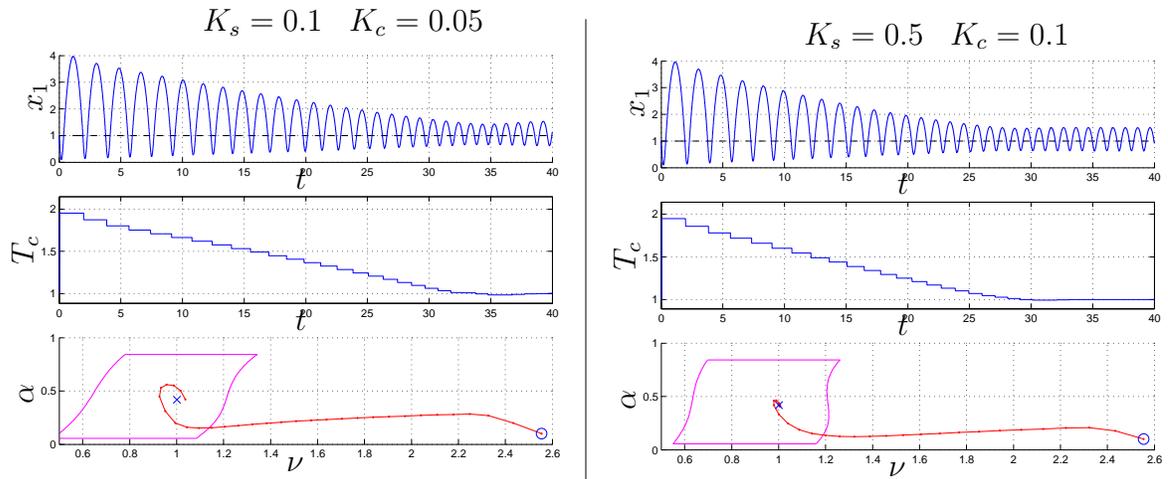


Figure 3.12: The Poincaré state flows of the closed-loop system that employs the hybrid tuning law in (3.22). Two different gain settings are exhibited. In the top figure the synchronizer gain,  $K_s = 0.1$ , is small, and therefore, the composition gain has to be set to a small value,  $K_c = 0.05$ . In the bottom plot a larger synchronization gain,  $K_s = 0.5$ , allows for a higher composition gain,  $K_c = 0.1$ . Recall composition gain,  $K_c$ , determines the speed of convergence.

## CHAPTER 4

### Conclusion

Legged animals outperform any man-made vehicle in natural settings. Therefore, it is generally accepted that legs present a better solution to negotiate unstructured and unstable terrain [30]. Unfortunately, the control of legged locomotion presents a very hard engineering problem. A typical legged locomotor is a highly nonlinear dynamical system which offers only limited and intermittent control authority over the motion of its center of mass. At first glance, the very nature of legged systems suggests that the control of legged locomotion must be strongly sensor driven in order to synchronize the control actions with the motion of the body.

Recent collaborations between biologists and engineers [2, 3] have identified a set of novel principles for legged locomotion control. RHex [9] — a highly dexterous and durable autonomous hexapod robot — is an embodiment of these new approaches and presents a prime example showing their strength. The most surprising aspect of RHex is that its locomotion controller is task-level open-loop. Consequently, the mechanical, electrical and algorithmic design of RHex is significantly simpler than most other robotic platforms [6]. In fact, this very simplicity of RHex is the basis of its success as a physical legged locomotion platform.

However, the design of open-loop controllers for legged locomotion remains pri-

marily an intuition-driven process that requires many iterations of trial and error [8]. This thesis attempts the beginnings of a formal framework to design and verify open-loop controllers for legged locomotion. Our ultimate goal is to provide designers the ability to construct open-loop controllers with guaranteed properties.

Physically relevant legged machines, such as RHex, are highly nonlinear and incorporate many degrees of freedom and, therefore, do not offer analytic tractability that is necessary to extract basic results. Therefore, in this preliminary work we choose to limit our attention to a much simpler, but illustrative, dynamical model, a clock driven 1-DOF hopper that we introduced in Chapter 2. This model contains in greatly abbreviated form the basic components of an open-loop controlled legged machine. The controller takes the form of a signal generator. The mechanical hopper presents a hybrid dynamical system with intermittent controllability. The controller modulates the spring stiffness in a periodic fashion as it is described by a profile, the shape function,  $\psi(\theta)$ .

In the previous literature on open-loop control of hopping [117,118], the excitation signal is limited to a pulse train. In contrast, our novel approach in Chapter 2 allows us to investigate arbitrary piece-wise excitation signals. This contributes greater understanding of the importance of the shape of the excitation signal in relation to the stability properties of the forward coupled system.

The central result of the analysis in Chapter 2 is a set of sufficient conditions that lead to stable behavior for the clock driven 1-DOF hopper. In fact, our numerical studies suggest that these conditions may in fact be necessary as well as sufficient. An important observation from this result is the key role of mechanical dissipation in achieving stable behaviors. More specifically, it follows from Theorem 1 that the coupled system can exhibit asymptotically stable behavior, if the mechanical system is lossy,  $\mu > 0$ . Our numerical bifurcation studies suggest that in the absence of

mechanical losses,  $\mu = 0$ , the coupled system is only neutrally stable. Previous work on clock driven hoppers [117, 118] has demonstrated that open-loop control of dissipative systems can lead to asymptotic behaviors, however, to our knowledge none of these studies have identified dissipation as a necessary component.

Leveraging the analytical insight from Chapter 2 we introduced a family of discrete-time feedback mechanisms in Chapter 3 which constitutes the second contribution of the thesis. These controllers augment the clock controller and update its parameters, the clock period,  $T_c$ , and the shape configuration,  $\mathbf{o}$ , once for each cycle of the clock. The discrete-time nature of our approach leads to a low effective sensor bandwidth utilization (compared to continuous-time feedback systems as in [55]). In order to emphasize this low sensory bandwidth utilization we term these controllers “tuning laws.” Our analysis of the closed-loop system has led to conditions on the tuning law that give rise to improvement in the coupled behavior relative to the open-loop case. Our criteria for the comparison of stability properties address two features of the limit behavior: the size of the basin of attraction; and the time constant of the local linearized system.

In essence, a tuning law is a map from the Poincaré states,  $(\nu, \alpha)$ , to the clock period,  $T_c$ . Note that the delay state,  $\alpha$ , is the time between the touchdown and clock reset. The touchdown speed,  $\nu$ , can be estimated as a function of the flight time,  $T_A$ , which is the time between the lift-off and touchdown events. Hence, a physical implementation of a tuning law only requires detection of the touchdown and the lift-off events which defines a simple sensor modality that maintains the design simplicity that the underlying open-loop controller enjoyed.

The basic results from Chapter 2 can also serve as guidelines for open-loop controller design. In Appendix B we present an algorithm to design open-loop controller parameters, clock period,  $T_c$ , and the shape configuration,  $\mathbf{o}$ , such that a user specified

mechanical hopping task is rendered locally asymptotically stable. In this preliminary study the design process does not enjoy absolute guarantees of stability, giving rise to a limit behavior that is close to but not necessarily the same as what the user-specifies. However, our numerical studies suggest that for a fairly wide spectrum of hopping tasks the algorithm successfully constructs controller parameters that yield limit behaviors close to the user-specified behavior.

Physically relevant mechanical systems tend to have highly nonlinear compliant members that interact with the environment through complex kinematic structures. Obviously, the linear prismatic hopper model is not a good representation of such physical systems. In Appendix C we present a numerical study that aims to demonstrate that the basic concepts we discovered in Chapter 2 persist in a wide family of clock driven 1-DOF mechanical models. An important observation in this study is the role of spring dynamics in the stability of the coupled system. We discovered that the the stiffness characteristics of a spring largely determine the properties of the stable limit behavior. Driven by this observation we categorize the spring laws into three groups: softening springs; linear springs; and hardening springs. Notice that the hopper model is a member of the softening spring class by the virtue of the gravitational potential.

Our numerical studies suggest that all cases are neutrally stable in the lossless setting, which matches our earlier observation in the hopper model. In lossy settings we observe that linear spring becomes unstable, whereas, both softening and hardening springs can exhibit locally asymptotically stable limit behaviors. However, the characteristics of the stable limit behavior differ for hardening and softening springs. For softening springs the stable operating regimes are those where the modulation of the spring parameter occurs as the spring goes through a compression followed by a decompression cycle. On the other hand, for hardening springs the stable oper-

ating regimes are those where the modulation occurs as the spring goes through a decompression followed by a compression.

An immediate implication of this observation is that there are certain spring types that are suitable for certain tasks. For instance, in a hopper the leg goes through a compression and decompression cycle during stance while the controller have authority over the mechanical system. According to our observations from above, a leg with a softening type spring is more suitable for open-loop control of hopping. Our numerical results agree with this observation. On the other hand, a hardening spring is better suited for a climbing task where the leg/arm goes through a decompression followed by a compression.

In the future work we plan to extend these concepts to open-loop control of high degree of freedom systems such as versions of the spring loaded inverted pendulum (SLIP) model [98]. Our working hypothesis is that a high degree of freedom cyclic mechanical system can be thought of as a network of coupled second order clocks. Under certain conditions the couplings in this network define a predominantly feed-forward coupling among the clocks, allowing us to apply the results from this study to identify conditions for stability.

## APPENDICES

# APPENDIX A

## Dimensionless Physical Coordinate System

The mechanical hopper dynamics in terms of the physical length (in meters),  $h \in \mathbb{R}^+$ , and physical time (in seconds),  $\tau \in \mathbb{R}^+$ , may be written as

$$\begin{aligned} \frac{d^2 h}{d\tau^2} &= \frac{1}{m} [-k(h - h_0) - b\frac{dh}{d\tau}], & h \in (0, h_0] \\ \frac{d^2 h}{d\tau^2} &= -g, & h \in (h_0, \infty). \end{aligned} \tag{A.1}$$

It is parameterized by five physical properties: the body mass (kg),  $m$ ; the leg spring stiffness (N/m),  $k$ ; leg damping (Ns/m),  $b$ ; leg rest length (m),  $h_0$ ; and gravitational acceleration ( $\text{m/s}^2$ ),  $g$ .

Due to the parsimonious representation it gives rise to we prefer to study the mechanical hopper in a dimensionless setup — composed of a dimensionless coordinate space,  $\mathcal{X} \subset \mathbb{R}^2$ , and dimensionless time,  $t \in \mathbb{R}^+$  — where the dynamics, given in (2.3) and (2.4), are parametrized by only two parameters: dimensionless leg stiffness,  $\xi$ ; and dimensionless leg damping,  $\mu$ .

The physical and dimensionless variables are related by two maps: a linear time transformation,

$$t = p_3\tau;$$

and, an affine length transformation,

$$x_1 = p_1(h + p_2).$$

whose parameters,  $(p_1, p_2, p_3)$ , are chosen so that the dynamical equations in terms of the dimensionless variables take the desired forms in (2.3) and (2.4).

By appropriately substituting the dimensionless expressions into (A.1) according to the relationships for speed,

$$\frac{dh}{d\tau} = \left( \frac{p_1}{p_3} \right) \frac{dx_1}{dt},$$

and acceleration,

$$\frac{d^2h}{d\tau^2} = \left( \frac{p_1}{p_3^2} \right) \frac{d^2x_1}{dt^2},$$

we rewrite the hopper dynamics in terms of the dimensionless variables,  $x_1$  and  $t$ , and the parameters of the two maps,  $(p_1, p_2, p_3)$ ,

$$\begin{aligned} \frac{d^2x_1}{dt^2} &= -\frac{k}{mp_3^2}x_1 + \frac{kp_1}{mp_3^2}(p_2 + h_0) - \frac{b}{mp_3} \frac{dx_1}{dt}, & p_1p_3 < h \leq p_1(h_0 + p_2), \\ \frac{d^2h}{d\tau^2} &= -\frac{p_1}{p_3^2}g, & p_1(h_0 + p_2) < h < \infty. \end{aligned} \quad (\text{A.2})$$

Simple comparison of the differential equations in (A.2) with those in (2.3) and (2.4) yields the following set of equations,

$$p_1p_2 = -1, \quad p_2 + h_0 = 0, \quad \frac{p_1}{p_3^2}g = 1,$$

whose unique solution,

$$p_1 = 1/h_0, \quad p_2 = -h_0, \quad p_3 = \sqrt{\frac{g}{h_0}},$$

results in the desired representations for the hopper dynamics in dimensionless coordinates,  $\mathcal{X}$ , given in (2.3) and (2.4) where the dimensionless system parameters are defined as

$$\xi := \sqrt{\frac{k}{mp_3^2}}, \quad \mu := \frac{b}{2mp_3}.$$

## APPENDIX B

### Open-Loop Controller Design

This chapter will discuss how to design open-loop controllers for 1-DOF hopping tasks. We will start with the task specification in Section B.1 which will be followed by a generic discussion of the design problem and the outline of a particular design process in Section B.2. In Section B.3 we will point out several key properties of the forcing function,  $\rho(\alpha)$  — a key player in the return map. We will consider a simple illustrative case study in Section B.4 to demonstrate the open-loop controller design. In more generic operating regimes the analytic structure of the the forcing function,  $\rho(\alpha)$ , and the remaining time function,  $\varphi_e(\alpha)$ , quickly becomes very complex. In order to gain more insight as to how the shape configuration,  $\mathbf{o}$ , affects these functions we will employ their approximates introduced in Section B.5. Finally, we will describe a design algorithm in Section B.6 that employs these approximate expressions.

#### B.1 Task Specification

For the purposes of the design we will limit our attention to those operating regimes of the coupled system in Section 2.1.3 that are described by the fundamental repeatable mode sequences,  $\mathcal{GAG}(e, N)$ . Note that the behavioral repertoire of the

coupled system is a two-parameter family. One can effectively define a target behavior using the limit Poincaré states: the fixed point speed,  $\nu^*$ , characterizes the properties of the mechanical motion; and the fixed point delay,  $\alpha^*$ , captures the phase difference between the controller clock and the mechanical hopper.

In this design discussion, we will only describe the target mechanical behavior which we choose to specify by the (dimensionless) physical apex height,  $x_1^{apex}$ . It follows from the definitions in Section 2.2.2 that the corresponding “desired” fixed point speed state is given by

$$\nu^* := \sqrt{2x_1^{apex}}. \quad (\text{B.1})$$

Note that our behavioral specification does not include the target phase difference,  $\alpha^*$ . We acknowledge that the synchronization aspect of the target behavior may also be of importance in studies such as those concerning neural control. However, our immediate focus in this study is to achieve a certain mechanical motion — a choice motivated by the needs in locomotion control. Relaxing constraints on the desired fixed point delay,  $\alpha^*$ , offers considerable simplification. Therefore, in the rest of the discussions the fixed point delay,  $\alpha^*$ , will be treated as if it were yet another parameter of the coupled system. The design process will automatically select the fixed point delay,  $\alpha^*$ .

## B.2 The Design Problem

In the most generic sense, the design can be stated as a constrained nonlinear root finding problem,

$$\mathbf{r}(\mathbf{p}^*, \mathbf{o}, T_c) = 1, \quad |\lambda_{\pm}(\mathbf{p}^*, \mathbf{o}, T_c)| < 0, \quad \mathbf{p}^* \in \mathcal{P}(\sigma), \quad (\text{B.2})$$

where  $\mathbf{p}^*$  is the fixed point in the target mode sequence,  $\sigma = \mathcal{GAG}(e, N)$ , that corresponds to the user-specified (desired) behavior. It is possible to employ some numerical technique to solve for controller parameters, the shape configuration,  $\mathbf{o}$ , and clock period,  $T_c$ . However, identification of a feasible initial condition for the numeric iterations and testing if a desired operating regime can be stable are not trivial in this generic formalization. Moreover, such a purely numerical approach is far from offering any further insight that may prove useful in other instances.

Instead, we will present an investigation of the return map,  $\mathbf{r}(\mathbf{p})$ , in relation to the controller parameters,  $(\mathbf{o}, T_c)$ , and present a design algorithm — a moral equivalent of the root-finding problem in (B.2). The overarching goal of our discussions is to establish design guidelines for the design process which may find applications in more general settings as in Section C.2.

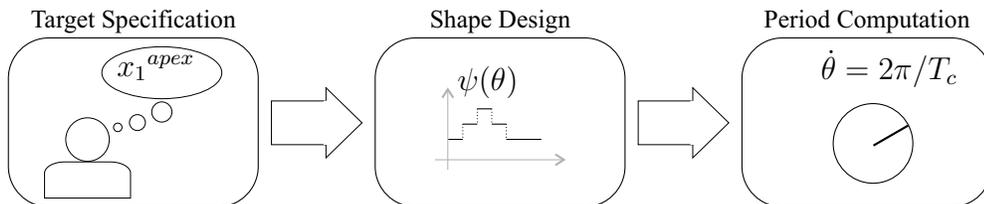


Figure B.1: The user specifies the desired behavior by the apex height of hopping,  $x_1^{apex}$ . The open-loop controller design is a two step process: 1) design of a “proper” shape function,  $\psi(\theta)$ ; and 2) computation of the clock period,  $T_c$ .

The special form of return map,  $\mathbf{r}(\mathbf{p})$ , in (2.22) allows us to compute the shape configuration,  $\mathbf{o}$  and the clock period,  $T_c$ , in a sequential manner. We first determine a “proper” shape configuration,  $\mathbf{o}$ , which in turn gives rise to a fixed point delay<sup>1</sup>,  $\alpha^*$ , according to the first equation in (2.33). We can directly compute the clock period,  $T_c$ ,

---

<sup>1</sup>Recall that the particular task description we employ in this discussion does not specify the fixed point delay,  $\alpha^*$ , which will be selected by the design process.

$$T_c(\alpha^*, \mathbf{o}) = 2\nu^* + \beta_{[1, e-1]} + \varphi_e(\alpha^*) + \alpha^* \quad (\text{B.3})$$

as a function of the selected shape configuration,  $\mathbf{o}$ , and the fixed point delay,  $\alpha$ , using the second equation in (2.33). In this process, depicted in Figure B.1, what constitutes as a proper shape configuration,  $\mathbf{o}$ , is the central question.

To address the issue of how to characterize proper shape configurations we will take advantage of the analytic studies in Chapter 2. For a given mechanical dissipation,  $(\mu, \zeta)$ , and a user-specified fixed point speed,  $\nu^*$ , the *sufficient* stability conditions in Theorem 1 define the “set of sufficient shapes,”  $\mathcal{O}_s(\mu, \zeta, \nu^*) \subset \mathcal{O}$ ,

$$\mathcal{O}_s(\mu, \zeta, \nu^*) := \{\mathcal{O} \mid \exists \alpha^* \in \mathcal{D}^*, \rho(\alpha^*) = 1 \wedge \rho'(\alpha^*) > 0 \wedge T_c(\nu^*, \alpha^*) \in \mathcal{T}_s(\alpha^*)\}, \quad (\text{B.4})$$

which by definition is a subset of the “set of stable shapes,”  $\mathcal{O}_*(\mu, \zeta, \nu^*)$  — the set of shape configurations where the target behavior is stable. In fact, our numerical studies suggests<sup>2</sup> that the set of sufficient shapes,  $\mathcal{O}_s(\mu, \zeta, \nu^*)$  is either identical or very close to the set of stable shapes,  $\mathcal{O}^*(\mu, \zeta, \nu^*)$ . Therefore, the conditions in Theorem 1 can effectively serve as design guidelines.

For our discussions we will pick a particular subset of the sufficient shape set,  $\mathcal{O}_s$ , by replacing the period condition,  $T_c(\nu^*, \alpha^*) \in \mathcal{T}_s(\alpha^*)$ , with a conservative but analytically simpler alternative,  $\text{tr}(\alpha^*, \nu^*) = 1 - \varphi_e'(\alpha^*) - 2\nu^*\rho'(\alpha^*) > 0$ . Note that all clock periods,  $T_c$ , that give rise to right half plane eigenvalues,  $\text{Re}[\lambda_{\pm}] = \text{tr}/2 > 0$ , are necessarily in the stable period interval,  $\mathcal{T}_s$ . The reader can refer to the root-locus analysis in Section 2.4.4 for the details. Furthermore, for any physically viable fixed point speed,  $\nu^* > 0$ , the formula in (B.3) yields a clock period,  $T_c$ , that is strictly

---

<sup>2</sup>In our numerical studies, we could not find any controller configuration,  $(\mathbf{o}, T_c)$ , where the coupled system in Section 2.1.3 is stable despite the (sufficient) stability conditions in Theorem 1 are not satisfied.

greater than the minimum admissible period,  $T_c^0$ , in (2.35). Hence, substituting this alternative conservative condition we define the “set of proper shape configurations,”  $\mathcal{O}_p(\mu, \zeta, \nu^*) \subset \mathcal{O}_s(\mu, \zeta, \nu^*)$ ,

$$\mathcal{O}_p(\mu, \zeta, \nu^*) := \left\{ \mathcal{O} \mid \exists \alpha^* \in \mathcal{D}^*, \rho(\alpha^*) = 1 \wedge \rho'(\alpha^*) \in \left( 0, \frac{1 - \varphi_e'(\alpha^*)}{2\nu^*} \right) \right\}, \quad (\text{B.5})$$

which is a subset of the sufficient shape configurations,  $\mathcal{O}_s$ . An instance of the shape function,  $\psi(\theta)$ , that is defined by a proper shape configuration,  $\mathbf{o} \in \mathcal{O}_p$ , will be referred as a “proper shape function.”

It directly follows that an open-loop controller with a proper shape function,  $\psi(\theta)$ , running at the period given by (B.3) renders the user-specified target behavior locally asymptotically stable. Note that we have effectively reduced the original design problem in (B.2) to the problem of finding a proper shape configuration,  $\mathbf{o} \in \mathcal{O}_p$ .

We observe that the duration of the lift-off cell,  $\beta_e$  and the duration of the last cell,  $\beta_N$ , — two entries of the shape configuration,  $\mathbf{o}$  — “exclusively” parameterize the domain of the invariant delay interval,  $\mathcal{D}^*$ , in (2.34) but they do not appear in the forcing function,  $\rho(\alpha)$ . Therefore, the domain of the invariant delay interval,  $\mathcal{D}^*$ , and the functional properties of the forcing function,  $\rho(\alpha)$ , are in fact mutually independent, and can be configured separately.

Inspired by this particular feature of the stability conditions we propose a three step process to compute proper shape configurations,  $\mathbf{o} \in \mathcal{O}_p$ . Figure B.2 depicts the three steps of the shape design. The first step computes a nominal leg stiffness,  $\xi_n$ , which defines a lower bound for the leg stiffness,  $\psi(\theta) \geq \xi_n$ . Note that the nominal stiffness must be sufficiently high,

$$\xi_n > \nu^*, \quad (\text{B.6})$$

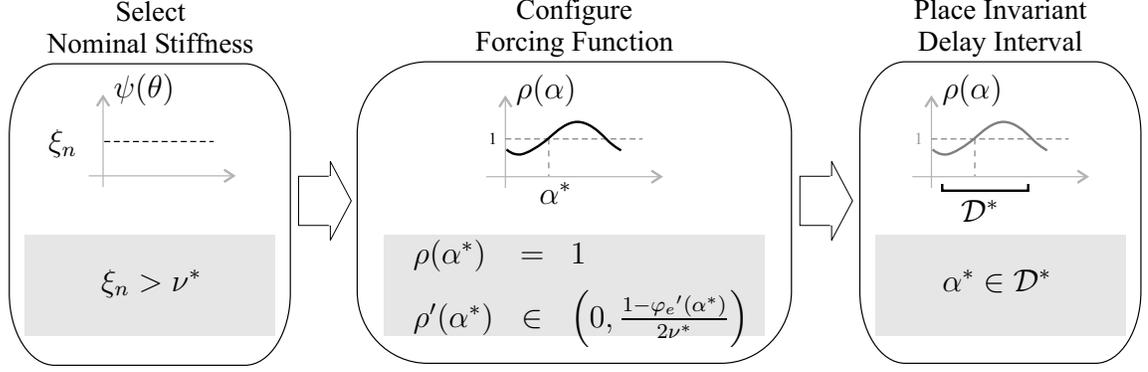


Figure B.2: The design of the shape function,  $\psi(\theta)$ , starts with selection of a nominal stiffness,  $\xi_n$ . Next, those entries of the shape configuration,  $\mathbf{o}$ , are computed such that the forcing function,  $\rho(\alpha)$ , intersects with the unity where its first derivative falls within the bounded interval in (B.5). Finally, the durations of the lift-off cell,  $\beta_e$ , and the last cell,  $\beta_N$ , are determined such that the invariant delay space,  $\mathcal{D}^*$ , captures this unity crossing point.

so that at the target behavior mechanical hopper does not collide with the ground. Second step focuses on those entries of the shape configuration,  $\mathbf{o}$ , that parametrize the forcing function,  $\rho(\alpha)$ , and the remaining time function,  $\varphi_e(\alpha)$ . The goal of this step is to identify a shape configuration,  $\mathbf{o}$ , such that the forcing function,  $\rho(\alpha)$ , crosses unity where its first derivative remains in the proper bounds. In essence, the second step is a constrained non-linear root-finding problem,

$$\rho(\alpha^*, \mathbf{o}) = 1, \quad \rho'(\alpha^*, \mathbf{o}) \in \left(0, \frac{1 - \varphi_e'(\alpha^*)}{2\nu^*}\right) \quad (\text{B.7})$$

where we treat both the shape configuration,  $\mathbf{o}$ , and the fixed point delay,  $\alpha^*$ , as parameters to be computed. In the final step, we properly position the invariant delay interval,  $\mathcal{D}^*$ , to capture the fixed point delay,  $\alpha^*$  in order to render the resulting fixed point,  $\mathbf{p}^*$ , valid.

Those lift-off and last cell durations,  $\{\beta_e, \beta_N\}$ , that result in  $\mathcal{D}^*(\sigma) \equiv \{\alpha^*\}$  are the shortest durations for these cells. In fact, in this configuration the sum of all cell durations would be identical to the smallest admissible clock period,  $T_c^0$ . The clock period,  $T_c$ , given by (B.3) is guaranteed to be larger than the smallest admissible period,  $T_c^0$ . Hence, to conclude the controller design the lift-off and last cell durations,  $\{\beta_e, \beta_N\}$ , will be increased properly to match the sum of cell durations to the computed clock period,  $T_c$ . Note that the span of the invariant delay interval,  $\mathcal{D}^*$ , in (2.34) monotonically increases with both the lift-off cell duration,  $\beta_e$ , and the last cell duration,  $\beta_N$ , therefore, after this final correction of the cell durations the fixed point delay,  $\alpha^*$ , will remain in the invariant delay interval  $\mathcal{D}^*(\sigma)$ .

We will demonstrate how to perform the computation of the proper shape,  $\mathbf{o} \in \mathcal{O}_p$ , first in a simple case study in Section B.4 and then later for a more generic family of operating regimes in Section B.6.

## B.3 Properties of the Forcing Function, $\rho(\alpha)$

The properties of the forcing function,  $\rho(\alpha)$ , are strongly related to the stability of the coupled system according to Theorem 1. This section identifies several key properties of the forcing function,  $\rho(\alpha)$ , which will be heavily utilized in the later sections. We will separately study each multiplicative term of the forcing function.

### B.3.1 The Loss Term, $\epsilon(\alpha)$

The loss term,  $\epsilon(\alpha)$ , represents the total mechanical energy loss over one cycle. It is a smooth positive-definite scalar function. Although the loss term in (2.24) is parameterized by the user-defined shape configuration,  $\mathbf{o}$ , as well as the mechanical dissipation parameters, the viscous damping coefficient,  $\mu$ , and the plastic collision

restitution coefficient,  $\zeta$ , its range is primarily determined by the mechanical loss parameters which are typically not user-selectable. Hence, we will treat the loss term,  $\epsilon(\alpha)$ , as the *uncontrollable* portion of the forcing function,  $\rho(\alpha)$ .

To achieve a stable operating regime mechanical losses must be compensated for by the actions of the controller. To much or too little energy injection leads to unstable behavior. Therefore, the range of the loss term,  $\epsilon(\alpha)$ , which effectively identifies bounds on the actions of the controller, plays a crucial part in the design process. To facilitate our later discussions we derive conservative bounds on the range of the loss term in Lemma 9.

**Lemma 9.** *For lossy settings,  $\mu > 0$  and  $\zeta \in (0, 1)$ , the range of the loss term,  $\epsilon(\alpha)$ , is bounded,*

$$1 > \zeta \exp\left[-\frac{\mu\pi}{\gamma_{max}}\right] \geq \epsilon(\alpha) \geq \zeta \exp\left[-\frac{\mu\pi}{\gamma_{min}}\right], \quad (\text{B.8})$$

over the valid delay interval,  $\alpha \in \mathcal{D}(\sigma)$ , where

$$\begin{aligned} \gamma_{min} &:= \min\{\gamma_N, \gamma_1, \dots, \gamma_e\} \\ \gamma_{max} &:= \max\{\gamma_N, \gamma_1, \dots, \gamma_e\}. \end{aligned}$$

**PROOF.** We define “stance time” as the total time spent in ground contact from touchdown to the following lift-off and denote by  $t_{\mathcal{G}}$ . It directly follows its definition in (2.24) that the loss term,  $\epsilon(\alpha) = \zeta \exp[-\mu t_{\mathcal{G}}]$ , is a monotonically decreasing function of the stance time,  $t_{\mathcal{G}}$ . Hence, we will concentrate on the stance time,  $t_{\mathcal{G}}$ , to derive bounds on the loss term,  $\epsilon(\alpha)$ .

In the preferred mechanical coordinate system,  $\mathcal{E}_N$ , let  $(\phi_i, \phi_{i+1})$  be the mechanical phase interval swept during the  $i^{\text{th}}$  stance mode,  $\mathcal{G}_i$ . The time it takes to sweep this interval monotonically decreases with the associated natural oscillation frequency,  $\gamma_i$ .

Hence, if we set the  $i^{\text{th}}$  cell natural oscillation frequency to the minimum,  $\gamma_{min} \leq \gamma_i$ , sweeping the same interval,  $(\phi_i, \phi_{i+1})$ , would require a longer or equal cell duration. It follows that if the natural oscillation frequencies of all cells are set to the minimum one,  $\gamma_{min}$ , while maintaining the same mechanical phase intervals the resulting stance time defines an upper bound,  $t_{\mathcal{G}}^{max} = \pi/\gamma_{min} > t_{GndMode}$ . Similarly, we can obtain a lower bound for the stance time,  $t_{\mathcal{G}}^{min} = \pi/\gamma_{max} < t_{\mathcal{G}}$ . The bounds on the range of the loss term in (B.8) follows the monotonic dependence between the loss term,  $\epsilon(\alpha)$ , and the stance time,  $t_{\mathcal{G}}$ .

For lossy settings,  $\mu > 0$ , and  $\zeta \in (0, 1)$ , the upper bound is strictly smaller than unity restricting the loss term,  $\epsilon(\alpha)$ , below unity. The loss term,  $\epsilon(\alpha)$ , is positive definite for lossy setting by definition. Notice that we implicitly assume that the coupled system operates in the specified mode sequence,  $\sigma$ , therefore, the above bounds are only valid over the associated valid delay interval,  $\mathcal{D}(\sigma)$ .

□

### B.3.2 The Action Terms, $l_i(\alpha)$

In the forward coupled system defined in Section 2.1.3, the clock controller modulates the leg stiffness,  $\xi$ , according to a periodic excitation signal as prescribed by the shape function,  $\psi(\theta)$ . For computational reasons the analysis of Section 2.1.2 limits the definition of the shape function,  $\psi(\theta)$ , to piece-wise constant functions. The discontinuous jumps of the shape function,  $\psi(\theta)$ , will be referred as the controller “actions.” By definition, each action occurs at the onset of a clock cell. To help distinguish actions we will adopt a naming convention where the action that occurs at the onset of the  $i^{\text{th}}$  cell,  $\Theta_i$ , will be called the  $i^{\text{th}}$  action.

In the context of controller design we will be interested in only those actions

that occur during ground contact,  $\mathcal{G}$ , since they alter the total mechanical energy of the mechanical hopper. Specifically, in a fundamental repeatable mode sequence,  $\mathcal{GAG}(e, N)$ , actions from the 1<sup>st</sup> through  $e^{\text{th}}$  occur during the stance mode,  $\mathcal{G}$ . The  $i^{\text{th}}$  action term,  $l_i(\alpha)$ , in the forcing function,  $\rho(\alpha)$ , captures the change in the total mechanical energy as a result of  $i^{\text{th}}$  action. Although, in an  $N$ -Cell shape function,  $\psi(\theta)$ , there are  $N$  actions, the action terms,  $l_i(\alpha)$ , are only defined for those actions that occur in stance mode,  $\mathcal{G}$ .

The expression of the  $i^{\text{th}}$  action,  $l_i(\alpha)$ , in (2.25) is a wrapped form of the associated energy component of the relating transformation,  $\mathbf{a}_{i-1}^i(\phi)$ , for which Lemma 10 offers an alternative expression. This simpler expression helps us identify two fundamental properties of the action terms,  $l_i(\alpha)$ . In Corollary 3 we will show that all action terms,  $l_i(\alpha)$  are periodic with period  $\pi/\gamma_N$ . Next, in Corollary 4 will identify the range of an action term,  $l_i(\alpha)$ , in terms of the shape configuration,  $\mathbf{o}$ .

**Lemma 10.** *The energy component of a relating transformation,  $\pi_1 \circ \mathbf{h}_i^j(e) = \eta \mathbf{a}_i^j(\phi)$ , is a linear function of the energy,  $\eta$ , and is a  $\pi$ -periodic function of the mechanical phase,  $\phi$ , whose phase dependent multiplicative term is given by*

$$\mathbf{a}_i^j(\phi) = \sqrt{(1 - A_i^j) + A_i^j \cos(2\phi - 2\phi_{TD}^i)} \quad (\text{B.9})$$

where  $\phi_{TD}^i = \arctan(\mu/\gamma_i)$  is the mechanical phase along the touchdown boundary in the  $i^{\text{th}}$  energy-phase coordinate system,  $\mathcal{E}_i$ , and  $A_i^j$  is the amplitude coefficient given by

$$A_i^j := \frac{1}{2} \left[ 1 - \left( \frac{\gamma_j}{\gamma_i} \right)^2 \right]. \quad (\text{B.10})$$

**PROOF.** To obtain the above simplification of the energy component of the relating transformation,  $\mathbf{a}_i^j(\phi)$ , we prefer to utilize its square,  $[\mathbf{a}_i^j(\phi)]^2$ , which has a simpler

algebraic structure. Since  $a_i^j(\phi) \geq 0$  by definition, the results obtained for the square of  $a_i^j(\phi)$  can be easily extended to itself. Taking the square of  $a_i^j(\phi)$ , which was defined in (2.14), and collecting the terms involving  $2\phi$  we obtain a phase shifted sinusoid with a positive offset,

$$[a_i^j(\phi)]^2 = \frac{K^2}{2} (1 + M^2 + L^2) + \frac{K^2}{2} \sqrt{(1 + M^2 - L^2)^2 + (2LM)^2} \cos(2\phi - \arctan[\frac{2LM}{1+M^2-L^2}]),$$

where

$$K = \frac{\gamma_j \xi_i}{\gamma_i \xi_j}, \quad L = \frac{\gamma_i \xi_j^2}{\gamma_j \xi_i^2}, \quad M = \frac{\mu}{\gamma_j} \left[ 1 - \left( \frac{\xi_j}{\xi_i} \right)^2 \right].$$

The squared expression,  $[a_i^j(\phi)]^2$ , has no inflection points. Hence, its extrema, where its first derivative vanishes, define local minima and maxima. The derivative of the squared expression with respect to the mechanical phase,

$$\frac{d}{d\phi} \left[ (a_i^j(\phi))^2 \right] = -(1 + M^2 - L^2) \sin(2\phi) + 2LM \cos(2\phi)$$

crosses zero at

$$\phi_k = k\frac{\pi}{2} + \phi_{TD}^i, \quad k = 0, 1, \dots$$

Hence, the squared expression,  $[a_i^j(\phi)]^2$ , evaluated at these local extrema,  $\phi_k$ , give rise to upper and lower bounds for the squared expression,

$$(a_i^j(\phi_{TD}^i))^2 = 1 \quad \text{and} \quad (a_i^j(\phi_{TD}^i + \pi/2))^2 = \frac{\gamma_j^2}{\gamma_i^2}.$$

Note that the squared expression,  $[a_i^j(\phi)]^2$ , is an affine transformation of a phase sinusoid,  $\cos(2\phi - 2\phi_{TD}^i)$ . It directly follows that using the extrema points we just derived we can rewrite the squared expression in a simplified form,

$$[a_i^j(\phi)]^2 = \frac{1}{2} \left( 1 + \frac{\gamma_j^2}{\gamma_i^2} \right) + \frac{1}{2} \left( 1 - \frac{\gamma_j^2}{\gamma_i^2} \right) \cos [2(\phi - \phi_{TD}^i)].$$

Defining an amplitude constant in (B.10) we obtain the simplified expression in (B.9). □

The alternative expression of the energy component of the relating transformation in (B.9) gives rise to a corresponding alternative action term expression,

$$l_i(\alpha) = \sqrt{(1 - A_i) + A_i \cos (2\phi_i^*(\alpha) - 2\phi_{TD}^i)}, \quad (\text{B.11})$$

where we define a shorthand notation,  $A_i \equiv A_{i-1}^i$ , that will be employed hereafter. The following results in Corollary 3 and Corollary 4 are inspired by this alternative expression of the action term.

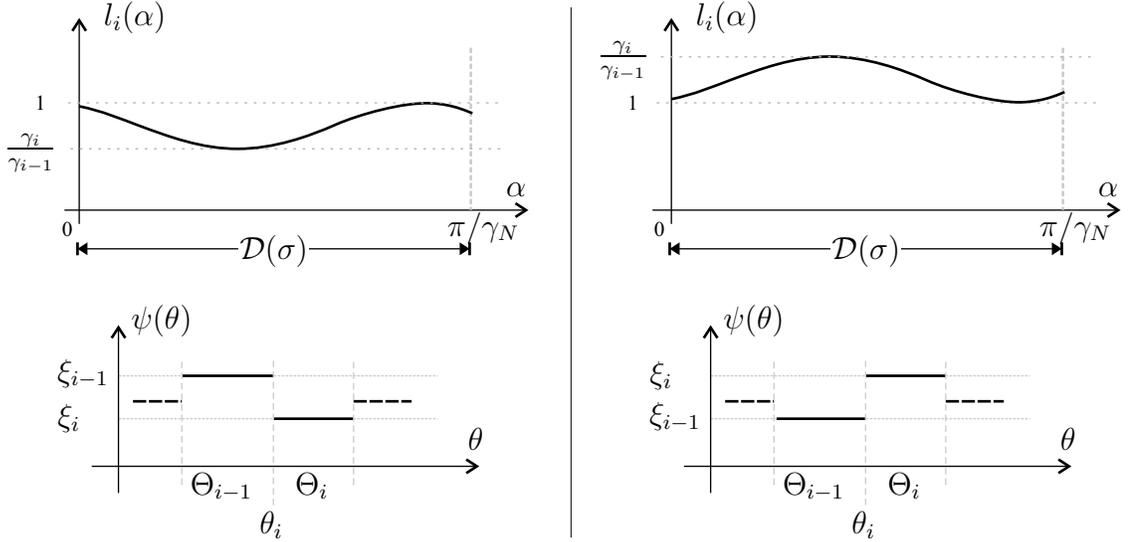


Figure B.3: Illustration of typical action term plots (top) and associated discontinuous change in the shape function,  $out(\theta)$  (bottom). The left side plots depict a relaxation action and the right side plots correspond to a stiffening actions.

According to the change in stiffness value we categorize actions into two groups: 1) “stiffening actions” where the stiffness increases,  $\xi_{i-1} < \xi_i$ ; and 2) “relaxation actions” where the stiffness decreases,  $\xi_{i-1} > \xi_i$ . Figure B.3 illustrates typical plots of the two types of action terms. Recall that action terms,  $l_i(\alpha)$ , appear as scaling terms in the energy map in (2.22). Their range, specified in Corollary 4, identify their energetic effects on the mechanical system. The stiffening actions inject energy into the mechanical system whereas relaxation actions cause a decrease in the total mechanical energy.

**Corollary 3.** *An action term,  $l_i(\alpha)$ , is a periodic function with period  $\pi/\gamma_N$ . It has no inflection points — points where both the first and the second derivatives vanish.*

PROOF. The switching phase at the  $i^{\text{th}}$  action,  $\phi_i^* : \mathcal{D} \rightarrow S^1$ , defined in (2.23), is a strictly monotonically increasing smooth function of the delay state,  $\alpha$ , and it is periodic with period  $2\pi/\gamma_N$ . It directly follows the definition in (2.25) and the alternative action term expression in (B.11) that action function is a positive definite periodic function of the delay,  $\alpha$ , with the period,  $\pi/\gamma_N$ .  $\square$

**Corollary 4.** *An action term,  $l_i(\alpha)$ , is a positive definite function whose range is the interval defined by unity and the ratio of the normalized frequencies,  $\gamma_i/\gamma_{i-1}$ . For a stiffening action,  $\xi_{i-1} < \xi_i$ , the action term,  $l_i(\alpha)$ , is larger than unity,*

$$\frac{\gamma_i}{\gamma_{i-1}} \geq l_i(\alpha) \geq 1 > 0.$$

*For a relaxation action,  $\xi_{i-1} > \xi_i$ , the action term,  $l_i(\alpha)$ , is less than unity,*

$$1 \geq l_i(\alpha) \geq \frac{\gamma_i}{\gamma_{i-1}} > 0.$$

PROOF. By definition in (2.25) the range of an action term,  $l_i(\alpha)$ , is identical to that of  $a_{i-1}^i(\phi)$ . Hence, it directly follows the result of Lemma 10 that an action term,  $l_i(\alpha)$ , is tightly bounded by the extrema values of the associated energy coefficient function,  $a_{i-1}^i(\phi)$ , which are the unity and the ratio of normalized oscillation frequencies,  $\gamma_i/\gamma_{i-1}$ . For a stiffening actions,  $\xi_{i-1} < \xi_i$ , the ratio of frequencies is larger than unity and restricts the action term above unity. Similarly, for a relaxation action,  $\xi_{i-1} > \xi_i$ , the action term is bounded below unity.

Since the switching phase function,  $\phi_i^*(\alpha)$ , is a diffeomorphism the action function,  $l_i(\alpha)$ , conserves the distribution of the extreme points of  $a_{i-1}^i(\phi)$  whose critical points contain minima and maxima, but no inflection points.

□

## B.4 An Illustrative Case Study

### B.4.1 The Single Action Mode Sequences

A “single action mode sequence,”  $\mathcal{GAG}(1, N)$ , is a fundamental repeatable mode sequence that has only a single action in the stance mode,  $\mathcal{G}$ . Figure B.4 illustrates a typical single action shape function,  $\psi(\theta)$ , and the progression of the states of the mechanical hopper between two consecutive Poincaré samples. In this illustrative case study we will limit our discussion to the physically relevant lossy mechanical setting,  $\mu > 0$  and  $\zeta \in (0, 1)$ , that is driven by a stiffening action. Using the results from Chapter 2 it can be easily shown that other settings are strictly unstable. Interestingly, the models defined by both Berkemeier [129] and Ringrose [128] operate in this particular single action mode sequence as well.

In the single action mode sequences,  $\mathcal{GAG}(1, N)$ , the hybrid structure of the cou-

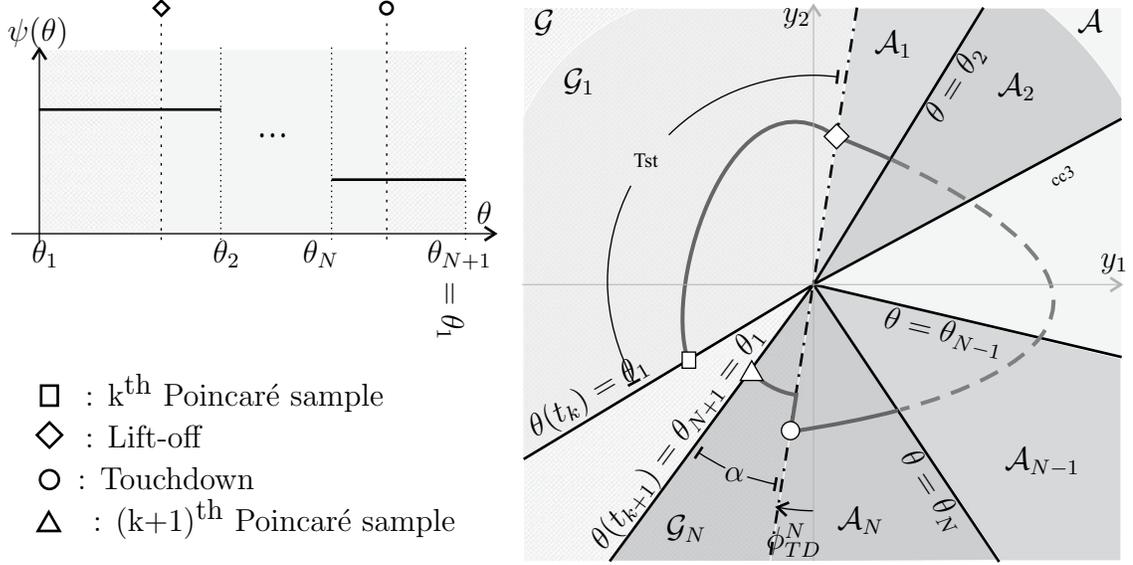


Figure B.4: A typical mechanical state trajectory during a single action operating regime between two consecutive Poincaré samples (right) and a typical single action shape function,  $\psi(\theta)$  (left). Crucial events along the trajectory are indicated by markers whose list can be found in the legend below.

pled dynamics assumes its simplest form. Naturally, this gives rise to a particularly simple return map,  $\mathbf{r}(\mathbf{p})$ . We will first identify several key properties of the single action return map,  $\mathbf{r}(\mathbf{p})$ . Our derivations will focus on the two key terms of the return map,  $\mathbf{r}(\mathbf{p})$ : 1) the single action remaining time function,  $\varphi_1(\alpha)$ ; and 2) the single action forcing function,  $\rho(\alpha)$ .

**Lemma 11.** *The single action remaining time function,  $\varphi_1(\alpha)$ , is a monotonically decreasing function whose only root is located at  $\alpha = \pi/\gamma_N$ .*

**PROOF.** The single action remaining time function,  $\varphi_1(\alpha)$ , is monotonically decreasing by definition. To derive its root we simply evaluate of its inverse at zero,  $\varphi_1^{-1}(0) = \mathbf{P}^{-1} \circ \mathbf{b}_1^N \circ \mathbf{R}_1(0)$ . The result follows from the direct algebraic manipulation

of the expression using Lemma 1.

□

**Lemma 12.** *The forcing function,  $\rho(\alpha)$ , of a single action fundamental repeatable mode sequence,  $\sigma = \mathcal{GAG}(1, N)$ , has three local extrema points within the valid delay interval,  $\mathcal{D}(\sigma) = \{0, \alpha_m, \pi/\gamma_N\}$ , and has no inflection points. If the action is of stiffening type than the extremum at  $\alpha = \alpha_m$  is a local maximum, otherwise, it is a local minimum.*

PROOF. The single action forcing function,  $\rho(\alpha) = \epsilon(\alpha)l_1(\alpha)$ , is composed of a loss term,  $\epsilon(\alpha)$ , and an action term,  $l_1(\alpha) = \sqrt{(1 - A_1) + A_1 \cos(2\gamma_N\alpha)}$ , which we choose to express in the alternative form defined in (B.11). The first derivative of the forcing function,

$$\rho'(\alpha) = \left( -A_1\epsilon(\alpha)[l_1(\alpha)]^{-2} \right) \left( \gamma_N \sin(2\gamma_N\alpha)l_1(\alpha) - \mu[1 - \cos(2\gamma_N\alpha)] \right).$$

has three roots,  $\{0, \alpha_m, \pi/\gamma_N\}$ , determined by the zero crossings of the second multiplicative term in its expression. By direct computation we show that the value of the second derivative of the forcing function,  $\rho''(\alpha)$ , evaluated at two outer most roots,  $\alpha \in \{0, \pi/\gamma_N\}$ ,

$$\rho''(\alpha = \{0, \pi/\gamma_N\}) = -2\gamma_N A_1 \epsilon(0) [l_1(0)]^{-2} \neq 0,$$

are identical and non-zero, and therefore, at  $\alpha \in \{0, \pi/\gamma_N\}$  the forcing function,  $\rho(\alpha)$ , has two local extrema of the same type — both are maxima and/or minima according to the sign of  $A_1$ . Since the forcing function,  $\rho(\alpha)$ , is continuous in delay,  $\alpha$ , it cannot have an inflection point at the intermediate root of the first derivative,

$\alpha_m \in (0, \pi/\gamma_N)$ , and must have a local extremum of the other kind. Hence, we conclude that the forcing function,  $\rho(\alpha)$ , has no inflection points.

For stiffening actions the amplitude coefficient is negative definite,  $A_1 < 0$ , and therefore, the local extrema at  $\alpha = \alpha_m$  is a local maxima where the forcing function takes its largest value over the valid delay interval,  $\mathcal{D}(\sigma)$ . Similarly, it can be shown that if the action is of relaxation type than the extrema at  $\alpha = \alpha_m$  is a local minima.

□

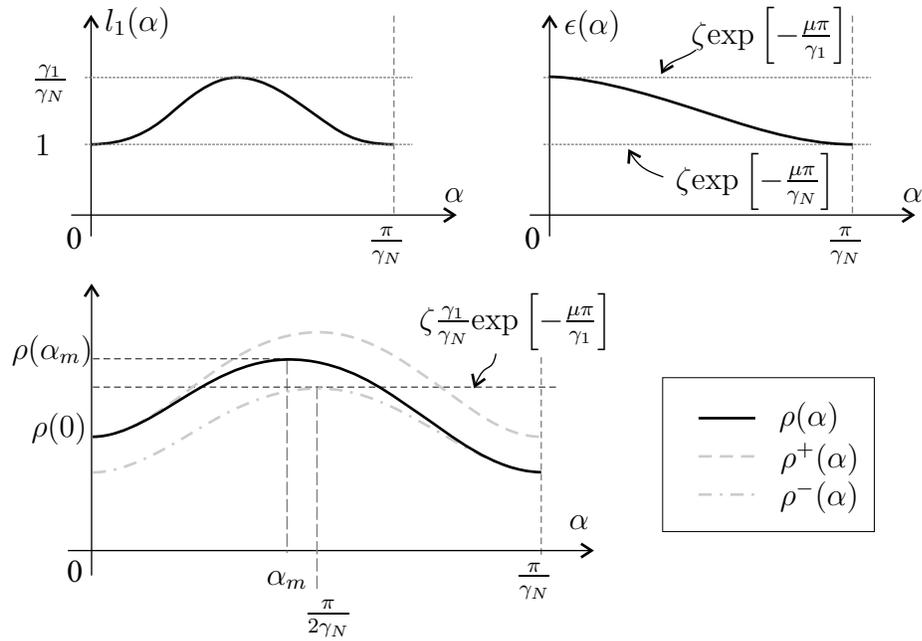


Figure B.5: The single action forcing function,  $\rho(\alpha)$ , and its bounds are depicted on the bottom sketch. The plots on the top illustrate the typical form of the single (stiffening) action,  $l_1(\alpha)$ , (left) and the single action loss term,  $\epsilon(\alpha)$  (right).

## B.4.2 Single Action Shape Design

This section describes a computational algorithm to design the shape function,  $\psi(\theta)$ , that is specifically tailored for the single action mode sequences,  $\mathcal{GAG}(1, N)$ . The general flow of the process was described in Section B.2.

The first step of the design is to pick a sufficiently large nominal stiffness according to (B.6) which we will use as the last cell stiffness,

$$\xi_N := \xi_n. \tag{B.12}$$

Note that only free parameter in both the single action forcing function,  $\rho(\alpha)$ , and the remaining time function,  $\varphi_e(\alpha)$ , is the lift-off cell stiffness,  $\xi_1$ , which we will identify next such that the single action forcing function,  $\rho(\alpha)$ , intersects with unity at a point where its first derivative falls within the interval specified in (B.5).

It directly follows from Lemma 12 that the single action forcing function,  $\rho(\alpha)$ , is monotonically increasing over  $(0, \alpha_m)$ . The peak value of the forcing function,  $\rho(\alpha_m)$ , is a monotonic function of the difference between the lift-off and last cells,  $\xi_1 - \xi_N$ . Since the first derivative,  $\rho'(\alpha)$ , is a concave function spanning  $(0, \overline{\rho'})$ , the first derivative of the forcing function at the unity crossing,  $\rho'(\alpha^*)$ , can be made arbitrarily small. Hence, for any fixed point delay,  $\nu^*$ , and dissipation level,  $(\mu, \zeta)$ , there exists an interval of lift-off stiffness values,  $\xi_1$ , where the conditions on the forcing function,  $\rho(\alpha)$ , in (B.5) are satisfied.

Using the Intermediate Value Theorem [130] we can conclude that there exists a unity crossing,  $\alpha^*$ , in the interval  $(0, \alpha_m)$  if the image of this interval,  $(\rho(0), \rho(\alpha_m))$ , contains unity. A direct computation yields the single action forcing function evaluated at the origin,

$$\rho(0) = \epsilon(0)l_1(0) = \zeta \exp \left[ -\frac{\mu\pi}{\gamma_1} \right],$$

Unfortunately, a closed form expression for  $\rho(\alpha_m)$  is not available. Instead, we will derive a lower bound for this expression. The single action forcing function consists of two positive definite multiplicative terms: the loss term,  $\epsilon(\alpha)$ ; and the (stiffening) action,  $l_1(\alpha)$ . Replacing the loss term,  $\epsilon(\alpha)$ , with the bounds from Lemma 9 we obtain an upper bound,

$$\rho^+(\alpha) := \zeta \exp \left[ -\frac{\mu\pi}{\gamma_1} \right] l_1(\alpha), \quad (\text{B.13})$$

and a lower bound,

$$\rho^-(\alpha) := \zeta \exp \left[ -\frac{\mu\pi}{\gamma_N} \right] l_1(\alpha), \quad (\text{B.14})$$

for the single action forcing function,  $\rho(\alpha)$ , as depicted in Figure B.5. Since both the single action forcing function,  $\rho(\alpha)$ , and its lower bound,  $\rho^-(\alpha)$ , each have a single maximum in the valid delay interval,  $\mathcal{D}(\sigma)$ , the peak value of the single action forcing function,  $\rho(\alpha_m)$ , is strictly larger than that of the lower bound,

$$\rho(\alpha = \alpha_m) > \max_{\alpha \in \mathcal{D}(\sigma)} [\rho^-(\alpha)] = \zeta \frac{\gamma_1}{\gamma_N} \exp \left[ -\frac{\mu\pi}{\gamma_N} \right], \quad (\text{B.15})$$

which we can derive by substituting the action term,  $l_1(\alpha)$ , with its upper bound from Corollary 4. For lossy settings,  $\mu > 0$  and  $\zeta \in (0, 1)$ , the single action forcing function evaluated at the origin is strictly less than unity,  $\rho(0) < 1$ . Hence, it follows (B.15) that selecting the cell stiffnesses,  $\{\xi_1, \xi_N\}$ , such that

$$\zeta \frac{\gamma_1}{\gamma_N} \exp \left[ -\frac{\mu\pi}{\gamma_N} \right] > 1, \quad (\text{B.16})$$

pushes the peak of the single action forcing function above unity,  $\rho(\alpha_m) > 1$ .

We will derive an upper bound for the first derivative of the single action forcing function,  $\rho'(\alpha)$ , in (B.4.1). At the fixed point we can replace the action term with the inverse of the loss term,  $l_1(\alpha^*) = 1/\epsilon(\alpha^*)$ . Substituting each term with proper upper bound expressions we obtain an upper bound for the first derivative at the unity crossing,

$$\rho'(\alpha^*) < -A_1\gamma_N\zeta^2\exp\left[-\frac{2\mu\pi}{\gamma_1}\right], \quad (\text{B.17})$$

which leads to a conservative condition,

$$-A_1\gamma_N\zeta^2\exp\left[-\frac{2\mu\pi}{\gamma_1}\right] \leq \frac{1 - \frac{\gamma_N}{\gamma_1}}{2\nu^*}, \quad (\text{B.18})$$

that can replace the derivative condition in (B.5).

Simultaneous solution of the conservative conditions in (B.16) and (B.18) concludes the design of the stiffness values,  $(\xi_1, \xi_N)$ . Due to their conservative nature this system of inequalities has a non-empty set of solutions if the target fixed point speed,  $\nu^*$ , is sufficiently small. Our numerical studies suggest that a physically relevant and sizable interval of target speed values can be addressed by this computational process.

The last step of the shape design is to configure the invariant delay interval,  $\mathcal{D}^*$ , such that it contains the fixed point delay state,  $\alpha^*$ . It follows from Lemma 11 that the single action remaining time function,  $\varphi_1(\alpha)$ , is strictly positive definite over the delay space,  $\mathcal{D}$ , which results in a simpler expression for the invariant delay space,

$$\mathcal{D}^*(\sigma) := \left(\max\{0, \varphi_1^{-1}(\beta_1)\}, \min\{\beta_N, \pi/\gamma_N\}\right). \quad (\text{B.19})$$

We will configure the invariant delay space,  $\mathcal{D}^*$ , such that it includes the interval,  $(0, \alpha_m)$ , where the single action forcing function,  $\rho(\alpha)$ , was configured to cross unity in the previous step. Unfortunately, we do not have a closed form expression for

$\alpha_m$ . It can be shown that  $\alpha < \pi/(2\gamma_N)$ , which we will use instead as a conservative approximate. By direct computation it can be shown that if  $\beta_1 = \beta_N\gamma_N/\gamma_1$ , then  $\varphi_1^{-1}(\beta_1) = 0$ . Setting  $\beta_N = \pi/(2\gamma_N)$  concludes the configuration of the invariant delay interval,  $\mathcal{D}^*$ .

## B.5 Surrogate Expressions

The analytical structure of the forcing function,  $\rho(\alpha)$ , and the remaining time function,  $\varphi_e(\alpha)$ , becomes very complicated as the number of actions in the stance mode,  $e$ , increases. Limiting our attention to the small action mode sequences — an interesting and physically relevant sub-family of fundamental repeatable mode sequences defined in Section B.5.1 — we will introduce approximate expressions for the forcing function,  $\rho(\alpha)$ , and the remaining time function,  $\varphi_e(\alpha)$ . Section B.5.5 will numerically demonstrate that the approximations and the original expressions are close for a wide range of shape configurations.

### B.5.1 Small Action Operating Regimes

Theoretical studies often consider discontinuous variations of the command to ease analytic studies. Specifically, in [128, 129] the control is embodied in a single discrete change of the control signal which strongly resembles the operating regime in Section B.4.1. Another illustrative example can be found in [1] which implements the control in terms of two discontinuous changes.

However, in the physical world all changes are continuous. Hence, the relevance of a model with instantaneous jumps is often inversely proportional to the magnitude of these discontinuous variations. Ideally one likes to understand the behavior of the coupled system in (2.6) driven by any arbitrary *continuous* controller output,  $\psi_c \in$

$C^1[\Theta]$ . However, such an analysis is hard, if not impractical. Instead, our framework offers an approximate analysis where the continuous shape function of interest,  $\psi_c$ , is represented by a piece-wise constant approximation,  $\psi$ , that has arbitrarily many cells,  $N \in \mathbb{Z}$ . Naturally, as the number of cells,  $N$ , increases the magnitude of each discontinuous change decreases monotonically.

This feature inspires the definition of the “small action shape functions,” — an  $N$ -cell piece-wise constant shape function,  $\psi(\theta)$ , which satisfies two conditions: 1) its range (the leg stiffness) is much larger than the viscous damping,  $\xi \gg \mu$  for all  $i = 1, 2, \dots, N$ ; and 2) the difference between consecutive cell values is “small,”  $|\xi_i - \xi_{i-1}| \ll 1$  for all  $i = 1, 2, \dots, N$ .

A coupled system driven by a small action shape function,  $\psi(\theta)$ , will be said to operate in a “small action mode sequence.” In the discussions unless stated otherwise a small action mode sequence will be a fundamental repeatable mode sequence.

## B.5.2 Approximate Expressions

In the return map,  $\mathbf{r}(\mathbf{p})$ , the primary source of analytic complexity is the repetitive occurrence of the nonlinear phase component of the relating transformations,  $b_i^j(\phi)$ , whose elimination can lead to considerable simplifications. In fact, we notice that the phase component of the relating transformation,  $b_i^j(\phi)$ , resembles the identity map,  $id(\phi)$ , where the functional distance between the two,

$$\delta b_i^j = \|b_i^j - id(\phi)\|_\infty = \max_{\phi \in S^1} [|b_i^j(\phi) - \phi|], \quad (\text{B.20})$$

monotonically increases with the magnitude of the difference between related cell stiffnesses,  $|\xi_i - \xi_j|$ , and the ratio of the damping to the natural oscillation frequency at the sink cell,  $\mu/\gamma_j$ .

This directly follows that for relating transformations between consecutive cells,

this error is small,  $\delta b_{i-1}^i \ll 1$ , if the shaping function,  $\psi(\theta)$ , satisfies two conditions of the small action mode sequences: 1) the stiffness values are much larger than the damping,  $\xi_j \gg \mu$ ; and 2) the changes in stiffness between consecutive cells are small,  $|\xi_i - \xi_{i-1}| \ll 1$ . This observation is the fundamental motivation behind the following derivation of the approximate return map.

The expression in (B.20) will be called as the “error in the phase component approximation.” Substituting the phase component,  $b_i^j(\phi)$ , with its approximate,  $\hat{b}_i^j(\phi) := id(\phi)$ , we define a family of “approximate relating transformations,”

$$\hat{\mathbf{h}}_i^j(\mathbf{e}) := \begin{bmatrix} \eta \cdot \mathbf{a}_i^j(\phi) \\ id(\phi) \end{bmatrix}. \quad (\text{B.21})$$

The error in the approximate relating transformations for consecutive cells is also small,

$$\left\| \hat{\mathbf{h}}_{i-1}^i - \hat{\mathbf{h}}_{i-1}^i \right\|_{\infty} \ll 1,$$

since  $\delta b_{i-1}^i \ll 1$ . Replacing each relating transformation,  $\mathbf{h}_i^j(\mathbf{e})$ , with its approximate,  $\hat{\mathbf{h}}_i^j(\mathbf{e})$ , and following the procedure outlined in Proposition 1 we compute “approximate switching phases,”

$$\hat{\phi}_i^*(\alpha) := \begin{bmatrix} i-1 \\ \odot \quad M_j \\ j=1 \end{bmatrix} \circ P(\alpha), \quad (\text{B.22})$$

and the “approximate remaining time function,”

$$\hat{\varphi}_e(\alpha) := R_e \circ \hat{\phi}_e^*(\alpha). \quad (\text{B.23})$$

Note that both approximate switching phases,  $\hat{\phi}_i^*(\alpha)$ , and the approximate remaining time function,  $\hat{\varphi}_e(\alpha)$ , are affine maps.

Simply following the definition of the original loss term in (2.24) we derive the “approximate loss term,”

$$\hat{\epsilon}(\alpha) := \zeta \exp \left[ -\mu \left( \alpha + \beta_{[1,e-1]} + \hat{\varphi}(\alpha) \right) \right] \quad (\text{B.24})$$

which we choose to reorganize into a more useful form,

$$\hat{\epsilon}(\alpha) = C \exp \left[ -\mu \left( 1 - \frac{\gamma_N}{\gamma_e} \right) \alpha \right],$$

where

$$C := \zeta \exp \left[ -\mu \left[ \beta_{[1,e-1]} + \frac{1}{\gamma_e} \left( \phi_{LO}^e - \phi_{TD}^N - \sum_{k=1}^{e-1} \gamma_k \beta_k \right) \right] \right]$$

is a constant which is less than unity for all lossy settings,  $\mu > 0$  and  $\zeta \in (0, 1)$ .

Similarly, substituting the switching term,  $\phi_i^*(\alpha)$ , with its approximate,  $\hat{\phi}_i^*(\alpha)$ , in (2.25) leads to the “approximate action terms,”

$$\hat{l}_i(\alpha) := \sqrt{(1 - A_i) + A_i \cos(2\gamma_N \alpha + 2S_i)}, \quad (\text{B.25})$$

which is parametrized by the action amplitude coefficient,  $A_i := A_{i-1}^i$ , defined in (B.10), and the action phase shift,

$$S_i := \phi_{TD}^N - \phi_{TD}^{i-1} + \sum_{k=1}^{i-1} \gamma_k \beta_k. \quad (\text{B.26})$$

The “approximate forcing function” is the multiplication of the approximate loss and the approximate action terms,

$$\hat{\rho}(\alpha) := \hat{\epsilon}(\alpha) \left[ \prod_{j=1}^e \hat{l}_j(\alpha) \right]. \quad (\text{B.27})$$

### B.5.3 Basic Forcing Function, $\hat{\rho}(\alpha)$

We choose to partition the forcing function,  $\rho(\alpha)$ , into two multiplicative terms:

1) the (uncontrollable) loss term,  $\epsilon(\alpha)$ ; and 2) the “cumulative action term,”

$$L(\alpha) := \prod_{i=1}^e l_i(\alpha).$$

which constitutes the controllable portion of the forcing function,  $\rho(\alpha)$ .

Consider the approximate forcing function,  $\hat{\rho}(\alpha)$ , that we defined in Section B.5.2.

The approximate loss term,  $\hat{\epsilon}(\alpha)$ , in (B.24) already has a very simple form. On the other hand, the approximate cumulative action term,

$$\begin{aligned} \hat{L}(\alpha) &= \prod_{i=1}^e \hat{l}_i(\alpha) \\ &= \sqrt{\left[ \prod_{i=1}^e (1 - A_i) \right] \left[ \prod_{k=1}^e \left( 1 + \frac{A_k}{1 - A_k} \cos(2\gamma_N \alpha + 2S_k) \right) \right]} \\ &= \sqrt{\left[ \prod_{i=1}^e (1 - A_i) \right] \left[ 1 + \sum_{k=1}^e \left( \frac{A_k}{1 - A_k} \cos(2\gamma_N \alpha + 2S_k) \right) + \text{HOT}_e \right]} \end{aligned}$$

contains super harmonics<sup>3</sup>,

$$\text{HOT}_1 := 0$$

$$\text{HOT}_i := \text{HOT}_{i-1} + \left[ \frac{A_i}{1 - A_i} \cos(2\gamma_N \alpha + 2S_i) \right] \left[ \prod_{k=2}^i \left( 1 + \frac{A_k}{1 - A_k} \cos(2\gamma_N \alpha + 2S_k) \right) - 1 \right]$$

that complicate its analytic structure. In small action settings the super harmonics,  $\text{HOT}_e$  are negligible compared to the sum of sinusoids which allows us to discard them leading to the definition of the “basic cumulative action term,”

---

<sup>3</sup>These are the additive terms that are products of different combinations of sinusoids and therefore have smaller period.

$$\hat{L}(\alpha) := \sqrt{\left[ \prod_{i=1}^e (1 - A_i) \right] \left[ 1 + \sum_{i=1}^e \frac{A_i}{1 - A_i} \cos(2\gamma_N \alpha + 2S_i) \right]}. \quad (\text{B.28})$$

as a coarser approximation to the cumulative action term,  $L(\alpha)$ . We define the basic forcing function,  $\hat{\rho}(\alpha)$ , as the product of the approximate loss term,  $\hat{\epsilon}(\alpha)$ , and basic cumulative action term,  $\hat{L}(\alpha)$ ,

$$\hat{\rho}(\alpha) := \hat{\epsilon}(\alpha) \hat{L}(\alpha). \quad (\text{B.29})$$

### B.5.4 Analytic Bounds

This section will derive upper and lower analytic bounds that simultaneously confine the forcing function,  $\rho(\alpha)$ , and its approximate,  $\hat{\rho}(\alpha)$ . To this end we will first compute bounds for each multiplicative term of the forcing function in terms of their respective approximates. The bounds on the forcing function will simply follow the definition of the forcing function in (2.26). The primary utility of these bounds is to show the surrogate expressions in Section B.5.2 and Section B.5.3 are close to their original counterparts.

#### The Bounds on the Switching Phase, $\phi_i^*(\alpha)$

To aid our derivations we will first concentrate on the switching function,  $\phi_i^*(\alpha)$ . We observe that the error in the switching phase approximation plays a key role. Lemma 13 states that the switching phase approximation error is bound from above by  $\delta\phi_i^*$ .

**Lemma 13.** *The error in  $i^{\text{th}}$  switching phase approximation,  $\hat{\phi}_i^*$ , is bounded from above,*

$$\left\| \phi_i^* - \hat{\phi}_i^* \right\|_{\infty} \leq \delta \phi_i^* \quad \text{where} \quad \delta \phi_i^* := \sum_{j=1}^{i-1} \delta b_{j-1}^j. \quad (\text{B.30})$$

PROOF. Our derivations will make use of recursive definitions for the  $i^{\text{th}}$  switching phase,

$$\phi_i^*(\alpha) = M_{i-1} \circ b_{i-2}^{i-1} \circ \phi_{i-1}^*(\alpha), \quad \phi_1^*(\alpha) = P(\alpha).$$

and its approximate,

$$\hat{\phi}_i^*(\alpha) = M_{i-1} \circ \hat{\phi}_{i-1}^*(\alpha), \quad \hat{\phi}_1^*(\alpha) = P(\alpha).$$

We take the difference between the  $i^{\text{th}}$  switching phase,  $\phi_i^*(\alpha)$ , and its approximate,  $\hat{\phi}_i^*(\alpha)$ , where the translational map,  $M_{i-1}$ , simplifies out. We add and subtract the  $i^{\text{th}}$  switching phase,  $\phi_i^*(\alpha)$ , to the right hand side of the equation. Regrouping terms yields a recursive formula for the difference expression,

$$\begin{aligned} \phi_1^*(\alpha) - \hat{\phi}_1^*(\alpha) &\equiv 0. \\ \phi_i^*(\alpha) - \hat{\phi}_i^*(\alpha) &= (b_{i-2}^{i-1} - id) \circ \phi_{i-1}^*(\alpha) + \left( \phi_{i-1}^*(\alpha) - \hat{\phi}_{i-1}^*(\alpha) \right), \end{aligned}$$

Next, we take the absolute value of both sides of the equality and apply the triangular inequality theorem to the right hand side. Replacing the first additive term on the right-hand side with its upper bound,

$$|b_{i-2}^{i-1}(\phi) - id(\phi)| \leq \delta b_{i-2}^{i-1},$$

given by (B.20), leads to a recursive formula to compute a (conservative) upper bound for the magnitude of difference between the  $i^{\text{th}}$  switching phase and its approximate,

$$\left| \phi_i^*(\alpha) - \hat{\phi}_i^*(\alpha) \right| \leq \delta b_{i-2}^{i-1} + \left| \phi_{i-1}^*(\alpha) - \hat{\phi}_{i-1}^*(\alpha) \right|.$$

The result in (B.30) directly follows the definition of  $\infty$ -norm and recursive application of the above inequality.

□

### The Bounds on the Remaining Time Function, $\varphi_e(\alpha)$

The remaining time function,  $\varphi_e(\alpha)$ , represents the time spent in stance mode,  $\mathcal{G}$ , during the lift-off cell. It is related to mechanical phase where the last action occurs as stated in (2.29). In Lemma 14 will employ the result from Lemma 13 and this relationship to derive an upper bound for the error in the remaining time function approximation.

**Lemma 14.** *The error in the remaining time function is bounded from above,*

$$\|\varphi_e - \hat{\varphi}_e\|_\infty \leq \frac{\delta\phi_{e+1}^*}{\gamma_e}. \quad (\text{B.31})$$

**PROOF.** We compute the difference between remaining time function and its approximate,

$$\varphi_e(\alpha) - \hat{\varphi}_e(\alpha) = -\frac{1}{\gamma_e} \left[ (\mathbf{b}_{e-1}^e - id) \circ \phi_e^*(\alpha) + \left( \phi_e^*(\alpha) - \hat{\phi}_e^*(\alpha) \right) \right],$$

which we rearrange the same way as in Lemma 13. Taking the absolute value of both sides of the equation and application of the triangular inequality theorem lead to an upper bound for the magnitude of the difference,

$$|\varphi_e(\alpha) - \hat{\varphi}_e(\alpha)| \leq \frac{1}{\gamma_e} \left[ |(\mathbf{b}_{e-1}^e - id) \circ \phi_e^*(\alpha)| + \left| \phi_e^*(\alpha) - \hat{\phi}_e^*(\alpha) \right| \right],$$

which we simplify further by substituting the upper bounds for the first term,  $|\mathbf{b}_{e-1}^e(\phi) - id(\phi)| \leq$

$\delta b_{e-1}^e$ , given by the (B.20), and the second term,  $\left| \phi_e^*(\alpha) - \hat{\phi}_e^*(\alpha) \right| \leq \delta \phi_e^*$ , given by Lemma 13. The result in (B.31) directly follows the definition of the  $\infty$ -norm and the switching phase approximation error bound,  $\delta \phi_i^*$  in (B.30). □

**Corollary 5.** *The remaining time function,  $\varphi_e(\alpha)$ , is bounded by an upper,  $\varphi_e^+(\alpha) := \hat{\varphi}_e(\alpha) + \frac{\delta \phi_{e+1}^*}{\gamma_e}$ , and a lower,  $\varphi_e^-(\alpha) := \hat{\varphi}_e(\alpha) - \frac{\delta \phi_{e+1}^*}{\gamma_e}$ , in terms of its approximate,  $\hat{\varphi}_e(\alpha)$ .*

PROOF. The result follows from Lemma 14. □

### The Bounds on the Action Terms, $l_i(\alpha)$

The  $i^{\text{th}}$  action term,  $l_i(\alpha)$ , is related to the  $i^{\text{th}}$  switching phase,  $\phi_i^*(\alpha)$ , by (2.25). In Lemma 15 we will derive an upper,  $l_i^+(\alpha)$ , and a lower,  $l_i^-(\alpha)$ , bound for a given action term,  $l_i(\alpha)$ , in terms of its approximate,  $\hat{l}_i(\alpha)$ . It is important to note that both action bounds,  $l_i^+(\alpha)$  and  $l_i^-(\alpha)$ , are linear functions of the approximate action expressions,  $\hat{l}_i(\alpha)$ .

**Lemma 15.** *The  $i^{\text{th}}$  action term,  $l_i(\alpha)$ , is bounded by an upper,  $l_i^+(\alpha) := \sqrt{1 + Q_i} \hat{l}_i(\alpha)$ , and a lower,  $l_i^-(\alpha) := \sqrt{1 - Q_i} \hat{l}_i(\alpha)$ , bound that are parameterized by*

$$Q_i := \frac{|A_i|}{\min(l_i)} \sqrt{2 [1 - \cos(2\delta \phi_i^*)]} \quad \text{where} \quad \min(l_i) = \begin{cases} 1 & ; \quad \xi_i \geq \xi_{i-1} \\ \gamma_i / \gamma_{i-1} & ; \quad \xi_i < \xi_{i-1} \end{cases} .$$

PROOF. Note that both the action terms,  $l_i(\alpha)$ , and their approximates,  $\hat{l}_i(\alpha)$ , are positive definite. This allows us to use squares of these expressions, which have

simpler analytic forms, in our derivations.

Expressing the approximate switching phase,  $\hat{\phi}_i^*(\alpha)$ , in terms of the original,  $\phi_i^*(\alpha)$ , and an additive error,  $\text{err}(\alpha)$ ,

$$\hat{\phi}_i^*(\alpha) = \phi_i^*(\alpha) + \text{err}(\alpha).$$

We take the difference between the squares of the action term,  $l_i(\alpha)$ , and its approximation,  $\hat{l}_i(\alpha)$ , which simplifies to

$$[l_i(\alpha)]^2 - [\hat{l}_i(\alpha)]^2 = A_i \sqrt{2[1 - \cos(2\text{err}(\alpha))]} \cos \left[ 2\phi_i^*(\alpha) - 2\phi_{TD}^{i-1} - \arctan \left( \frac{\sin(2\text{err}(\alpha))}{1 - \cos(2\text{err}(\alpha))} \right) \right],$$

Next, we divide both sides of the above equation by the square of the approximate action function,  $[\hat{l}_i(\alpha)]^2$ , and take the square root of both sides of the equation which yields

$$\frac{l_i(\alpha)}{\hat{l}_i(\alpha)} = \sqrt{1 + \frac{A_i \sqrt{2[1 - \cos(2\text{err}(\alpha))]} \cos \left[ 2\phi_i^*(\alpha) - 2\phi_{TD}^{i-1} - \arctan \left( \frac{\sin(2\text{err}(\alpha))}{1 - \cos(2\text{err}(\alpha))} \right) \right]}{\hat{l}_i(\alpha)}}.$$

Now, we will focus on the second additive term inside the square-root on the right hand side of this equation and determine (conservative) upper and lower bounds for it which, in turn, will produce bounds for the ratio,  $l_i(\alpha)/\hat{l}_i(\alpha)$ . First, we will replace the positive definite denominator,  $\hat{l}_i(\alpha)$ , with its minimum value,  $\min [\hat{l}_i(\alpha)] = \min [l_i(\alpha)]$ , specified in Corollary 4. This maximizes the magnitude of the overall ratio. Next, we will identify upper and lower bounds for the numerator. The action amplitude,  $A_i$ , whose sign is indeterminate, will be replaced by its absolute value. The second (positive definite) term is monotonically related to the switching phase error,  $\text{err}(\alpha)$ , and therefore, will be substituted by its maximum value,  $\max [\text{err}(\alpha)] = \delta\phi_i^*$ ,

given in Lemma 13. We will replace the third term with its maximum, 1, to derive an upper bound for the action term,

$$l_i(\alpha) \leq \sqrt{1 + \frac{|A_i| \sqrt{2[1 - \cos(2\delta\phi_i^*)]}}{\min(l_i)}} \hat{l}_i(\alpha),$$

Similarly, replacing the same term with its minimum,  $-1$ , yields a lower bound for the action term,

$$l_i(\alpha) \geq \sqrt{1 - \frac{|A_i| \sqrt{2[1 - \cos(2\delta\phi_i^*)]}}{\min(l_i)}} \hat{l}_i(\alpha),$$

Note that both bounds are linear functions of the approximate action term,  $\hat{l}_i(\alpha)$ .

□

### The Bounds on the Loss Term, $\epsilon(\alpha)$

The monotonic relationship between the loss term,  $\epsilon(\alpha)$ , and the remaining time function,  $\varphi_e(\alpha)$ , is the basis of the derivations in this section. The remaining time function,  $\varphi_e(\alpha)$ , can be rewritten as the sum of its approximate,  $\hat{\varphi}_e(\alpha)$ , and an additive error term,  $\text{err}(\alpha)$ , which allows us to relate the loss term,  $\epsilon(\alpha)$ , to the approximate loss term,  $\hat{\epsilon}(\alpha)$ . In Lemma 16 we use this observation in conjunction with the conservative error bound for the remaining time function approximate from Lemma 14 to derive conservative upper and lower bounds for the loss term,  $\epsilon(\alpha)$ .

**Lemma 16.** *The loss term,  $\epsilon(\alpha)$ , is bounded by an upper,  $\epsilon^+(\alpha)$ , and a lower bound,  $\epsilon^-(\alpha)$ , where*

$$\epsilon^+(\alpha) := \exp\left[\frac{\mu}{\gamma_e} \delta\phi_{e+1}^*\right] \hat{\epsilon}(\alpha) \quad \text{and} \quad \epsilon^-(\alpha) := \exp\left[-\frac{\mu}{\gamma_e} \delta\phi_{e+1}^*\right] \hat{\epsilon}(\alpha). \quad (\text{B.32})$$

PROOF. Defining an error expression,  $\text{err}(\alpha) := \hat{\varphi}_e(\alpha) - \varphi_e(\alpha)$ , we establish a relationship between the loss term,  $\epsilon(\alpha)$ , and its approximate,  $\hat{\epsilon}(\alpha)$ , given by

$$\epsilon(\alpha) = \exp [\mu \text{err}(\alpha)] \hat{\epsilon}(\alpha).$$

The first multiplicative term on the right hand side of the equation is a monotonic function of the error,  $\text{err}(\alpha)$ , whose magnitude is bounded,  $|\text{err}(\alpha)| \leq \delta\phi_{e+1}^*/\gamma_e$ , according to Lemma 14. It directly follows that substituting the smallest error value leads to a lower bound for the loss term,

$$\epsilon(\alpha) \geq \exp \left[ -\frac{\mu}{\gamma_e} \delta\phi_{e+1}^* \right] \hat{\epsilon}(\alpha).$$

Similarly, largest error value yields an upper bound for the loss term,

$$\epsilon(\alpha) \leq \exp \left[ \frac{\mu}{\gamma_e} \delta\phi_{e+1}^* \right] \hat{\epsilon}(\alpha).$$

□

### The Bounds on the Forcing Function, $\rho(\alpha)$

Using the definition of the approximate forcing function,  $\hat{\rho}(\alpha)$ , in (B.27) and the results in Lemma 16 and Lemma 15 we derive upper,  $\rho^+(\alpha)$ , and lower,  $\rho^-(\alpha)$ , bounds for the forcing function,  $\rho(\alpha)$ , in terms of its approximate,  $\hat{\rho}(\alpha)$ , in Lemma 17.

**Lemma 17.** *The forcing function,  $\rho(\alpha)$ , is bounded by an upper,*

$$\rho^+(\alpha) := \exp \left[ \frac{\mu}{\gamma_e} \delta\phi_{e+1}^* \right] \left( \prod_{i=1}^e \sqrt{1 + Q_i} \right) \hat{\rho}(\alpha) \quad (\text{B.33})$$

*and a lower bound,*

$$\rho^-(\alpha) := \exp \left[ -\frac{\mu}{\gamma_e} \delta\phi_{e+1}^* \right] \left( \prod_{i=1}^e \sqrt{1 - Q_i} \right) \hat{\rho}(\alpha). \quad (\text{B.34})$$

PROOF. Multiplicative terms of the forcing function, loss,  $\epsilon(\alpha)$ , and actions,  $l_i(\alpha)$ , as well as their respective bounds are all positive definite. Therefore, replacing each multiplicative term with its upper bound from Lemma 16 and Lemma 15 leads to an upper bound for the forcing function,  $\rho(\alpha) \leq \rho^+(\alpha)$ , given in (B.33). Similarly, substitution of the lower bounds yields a lower bound for the forcing function,  $\rho(\alpha) \geq \rho^-(\alpha)$ , given in (B.34). □

### B.5.5 Numerical Verification

The forcing function,  $\rho(\alpha)$ , and its approximation,  $\hat{\rho}(\alpha)$ , are both confined in the space defined by the upper,  $\rho^+(\alpha)$ , and lower,  $\rho^-(\alpha)$ , forcing bounds. In small action settings the upper,  $\rho^+(\alpha)$ , and lower,  $\rho^-(\alpha)$ , are close. Our numerical studies suggest that the approximate forcing function,  $\hat{\rho}(\alpha)$ , strongly resembles the original forcing function,  $\rho(\alpha)$ , for a wide range of shape configurations. Figure B.6 depicts a triangle shape function,  $\psi(\theta)$ , which only loosely satisfy the conditions of the small action operating regimes. Yet, the approximate expressions for both the forcing function,  $\rho(\alpha)$ , and the remaining time function,  $\varphi_e(\alpha)$ , are well approximated by their approximates. Figure B.7 illustrates the approximation performance in a typical small action operating regime where the approximate and the original expressions are very close.

The basic forcing function,  $\hat{\rho}(\alpha)$ , defined in (B.29), is, in essence, an approximation to the approximate forcing function,  $\hat{\rho}(\alpha)$ , itself. Despite this additional level

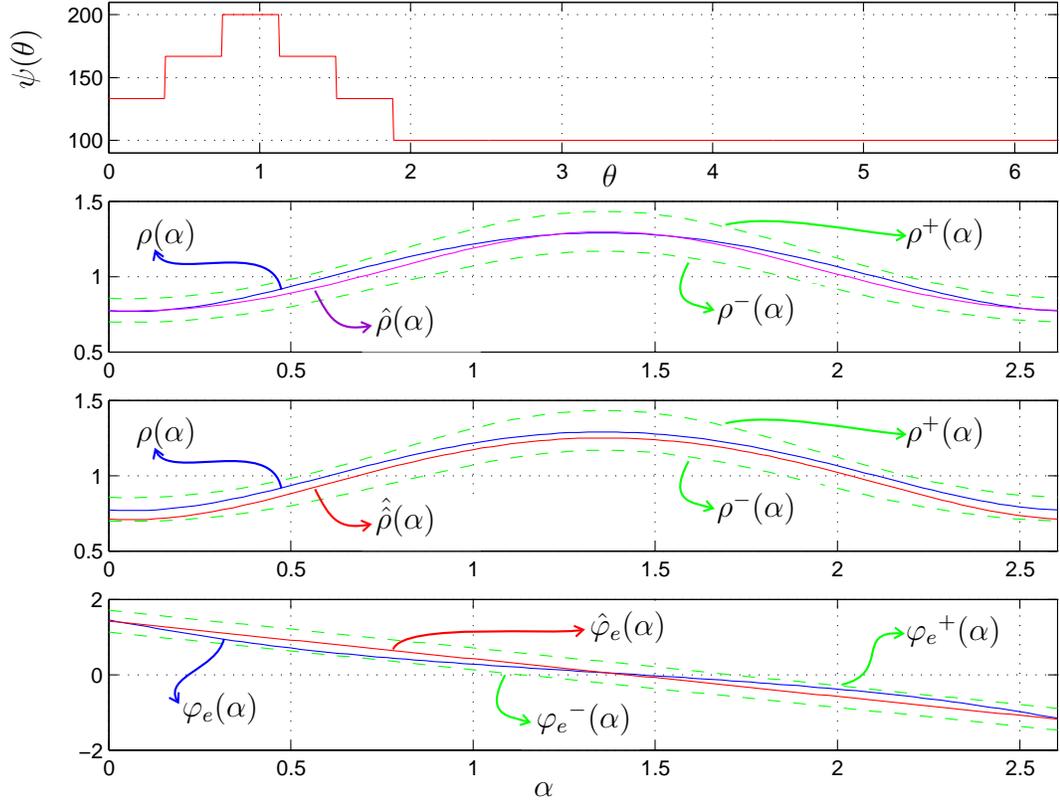


Figure B.6: Demonstration of the closeness between the approximate and original expressions of the forcing function and the remaining time function in an undesirable setting. Top shows a 4-cell triangular shape function,  $\psi(\theta)$ , for which the plots are generated. Middle two plots compare the original forcing function,  $\rho(\alpha)$ , to the approximate forcing function,  $\hat{\rho}(\alpha)$ , and the basic forcing function,  $\hat{\hat{\rho}}(\alpha)$ , along with the analytic bounds. The bottom plot compares the remaining time function,  $\varphi_e(\alpha)$ , to its approximation,  $\hat{\varphi}_e(\alpha)$ .

of simplification in the basic forcing expression, our numerical studies suggest that the basic forcing function,  $\hat{\hat{\rho}}(\alpha)$ , is also a good approximation of the original forcing function,  $\rho(\alpha)$ , over a fairly large portion of the shape configuration space,  $\mathcal{O}$ . Figure

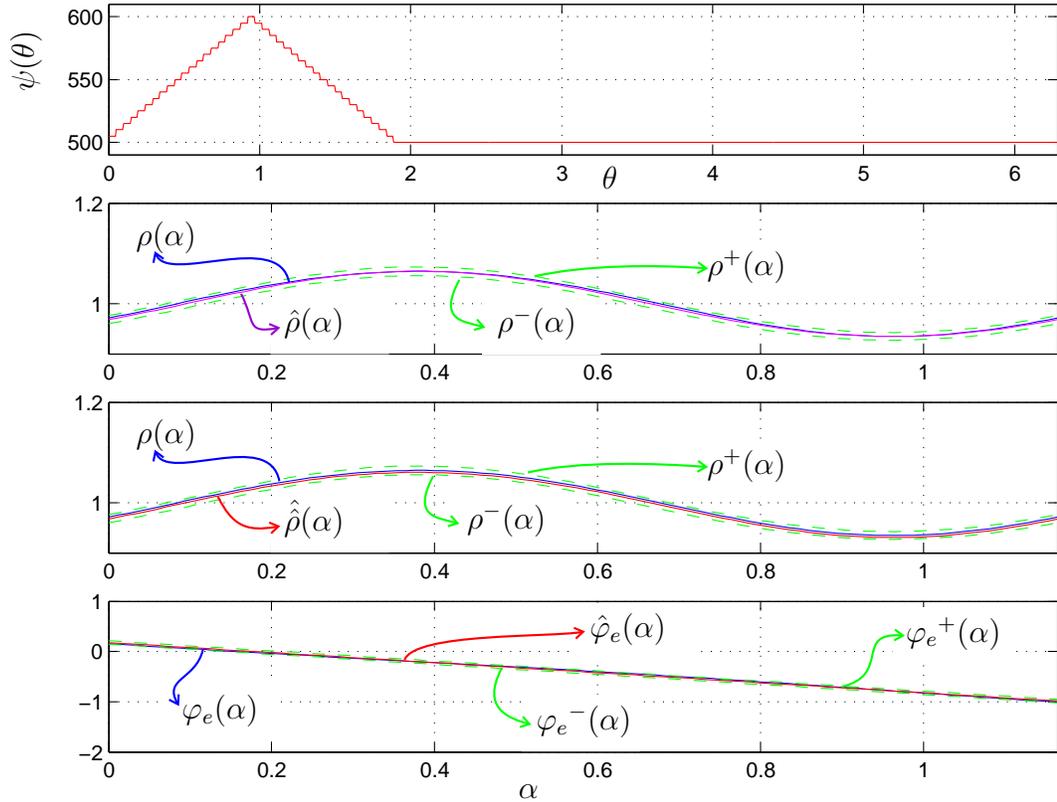


Figure B.7: Demonstration of the closeness between the approximate and original expressions of the forcing function and the remaining time function in a typical small action operating regime. Top shows a 40-cell triangular shape function,  $\psi(\theta)$ , for which the plots are generated. Middle two plots compare the original forcing function,  $\rho(\alpha)$ , to the approximate forcing function,  $\hat{\rho}(\alpha)$ , and the basic forcing function,  $\hat{\hat{\rho}}(\alpha)$ , along with the analytic bounds. The bottom plot compares the remaining time function,  $\varphi_e(\alpha)$ , to its approximation,  $\hat{\varphi}_e(\alpha)$ .

B.6 demonstrates a typical basic forcing function,  $\hat{\hat{\rho}}(\alpha)$ , in relation to the corresponding original forcing function,  $\rho(\alpha)$ , allowing the reader to assess the impact of the assumptions we made in the basic forcing function derivation in Section B.5.3.

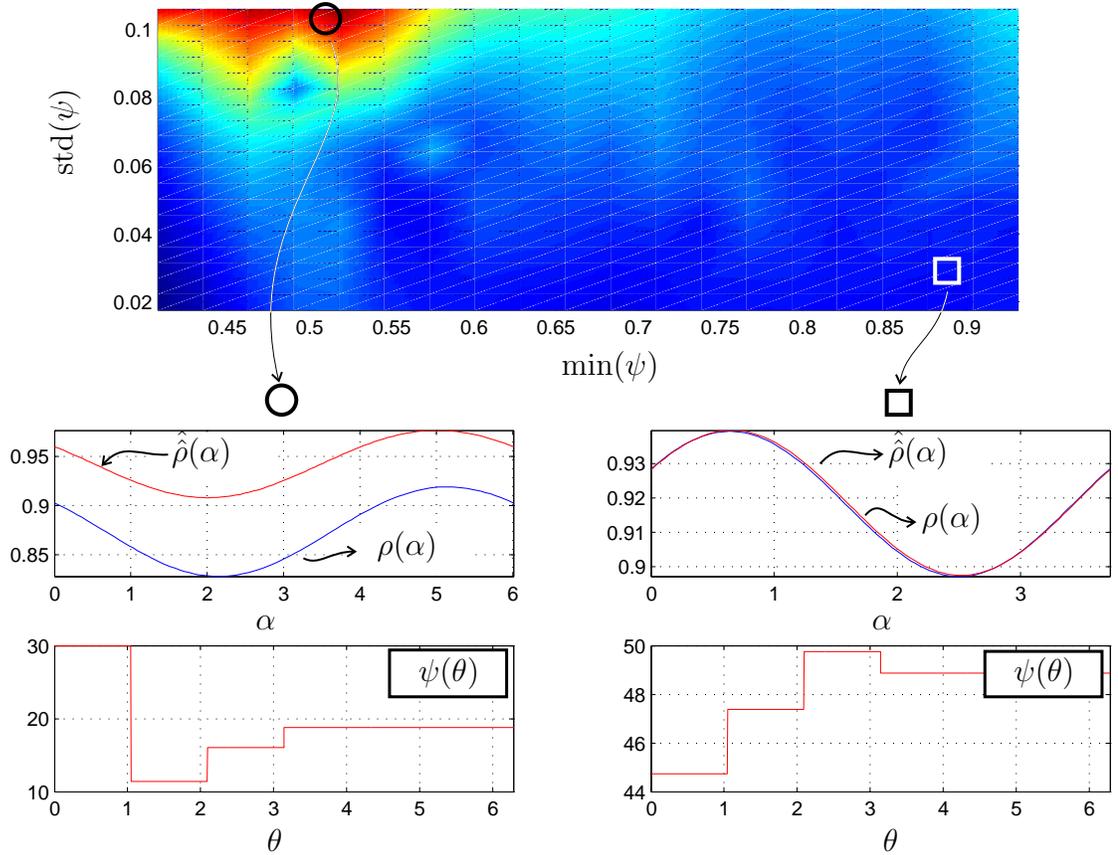


Figure B.8: The top plot illustrates numerically evaluated basic forcing error as a function of two properties of the shape function,  $\psi$ : its minimum,  $\min(\psi)$ ; and its standard deviation,  $\text{std}(\psi)$ . The magnitude of the error is color coded such that error grows from blue to red monotonically. The bottom plots demonstrate the shape functions,  $\psi(\theta)$ , and the comparison of the original forcing function,  $\rho(\alpha)$ , with the basic forcing function,  $\hat{\rho}(\alpha)$ . The pair on the left corresponds to the worst while the pair on the right depicts the best case observed in this study.

One major use of the small action shape functions is to approximate arbitrary continuously varying excitation profiles. Let  $\psi(\theta)$ , be a piece-wise constant approximation of a continuous profile,  $\psi_c(\theta)$ . As the number of cells of the piece-wise constant shape,  $N$ , increases the magnitude of the output change at every discontinuous jump

will decrease monotonically. Our numerical studies suggest that the basic forcing function,  $\hat{\rho}(\alpha)$ , does not diverge from the original as the number of cells,  $N$ , increases. We demonstrate this for three typical shape functions: a ramp function in Figure B.9; a triangle function in Figure B.10; and a half sinusoid in Figure B.11.

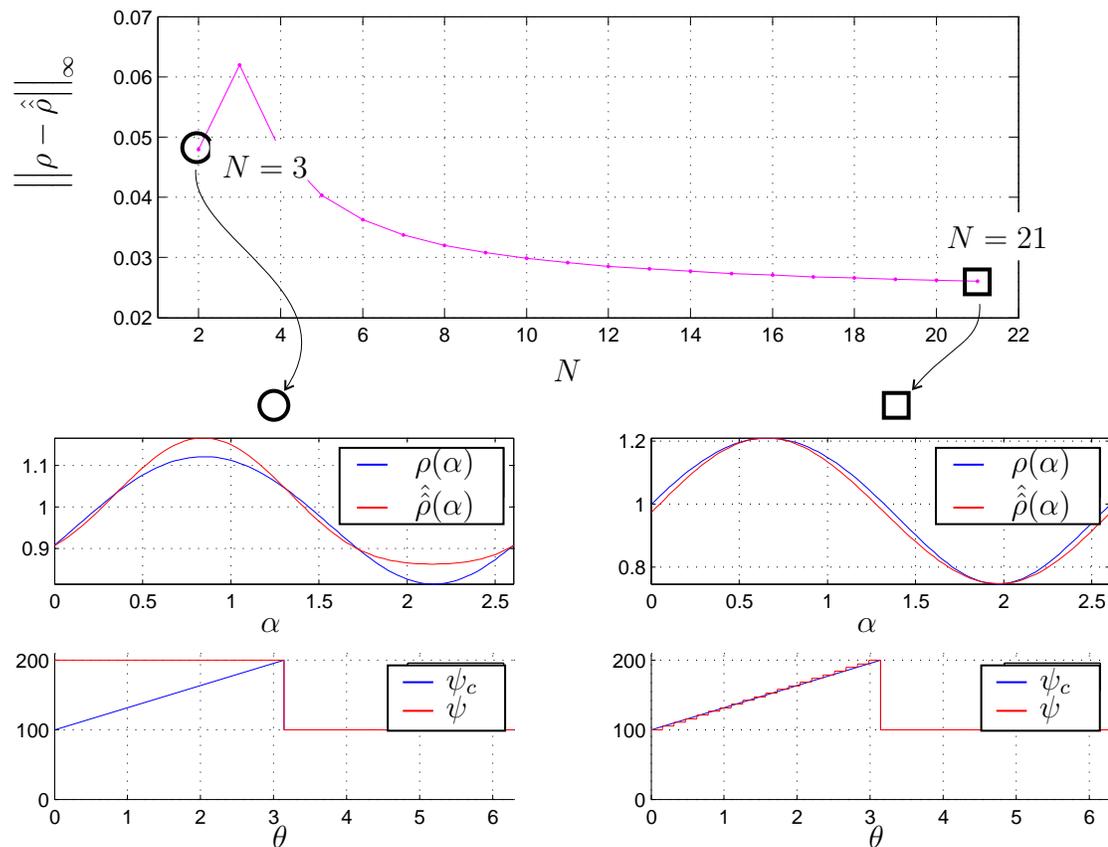


Figure B.9: The limit approximation error in the basic forcing function,  $\hat{\rho}(\alpha)$ , as the number of cells in the shape function,  $N$ , increases. The continuous shape template is a ramp function. The small action approximate is obtained by uniform sampling-and-hold.

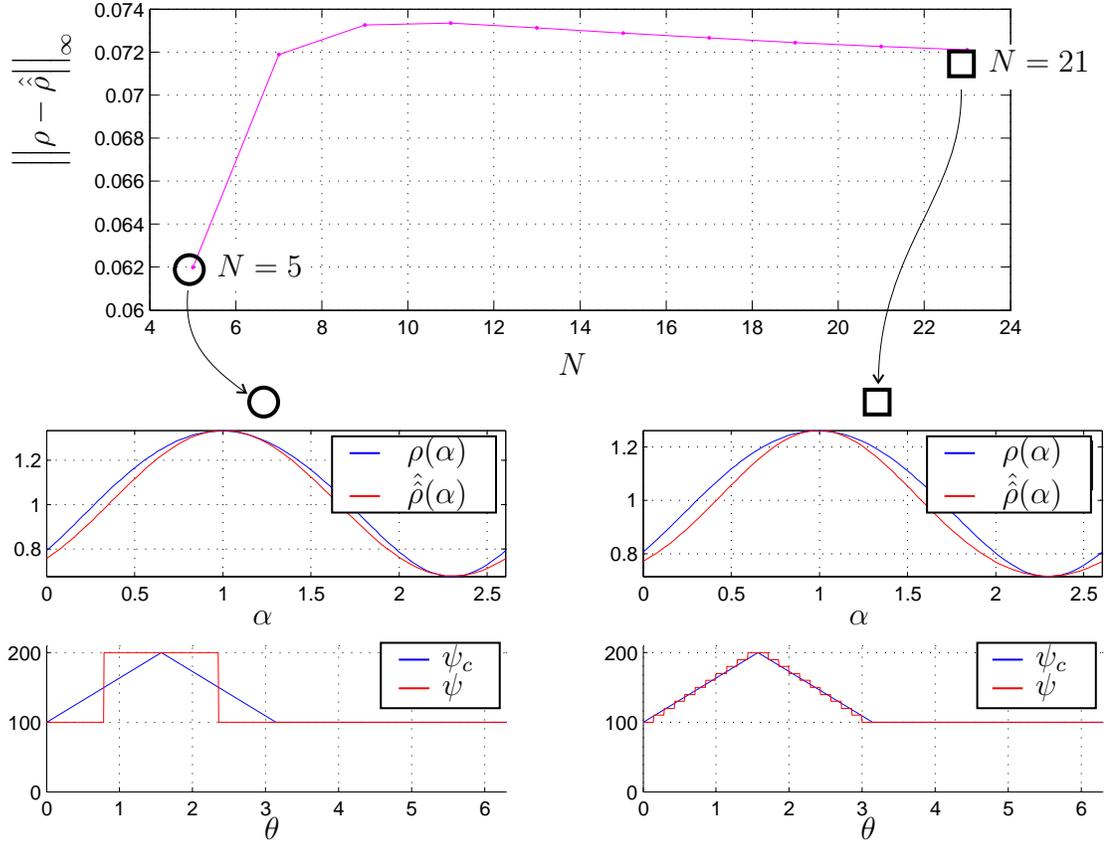


Figure B.10: The limit approximation error in the basic forcing function,  $\hat{\rho}(\alpha)$ , as the number of cells in the shape function,  $N$ , increases. The continuous shape template is a triangle function. The small action approximate is obtained by uniform sampling-and-hold.

## B.6 An Approximate Design Algorithm

This section will describe a procedure to design small action shape functions,  $\psi(\theta)$ , such that the (sufficient) stability conditions in Theorem 1 are met for a user-specified task. The basic structure of the procedure is identical to the generic design process described in Section B.2. However, in this procedure we will use the basic forcing function,  $\hat{\rho}(\alpha)$ , and approximate remaining time function,  $\hat{\varphi}_e(\alpha)$ , from Section B.5

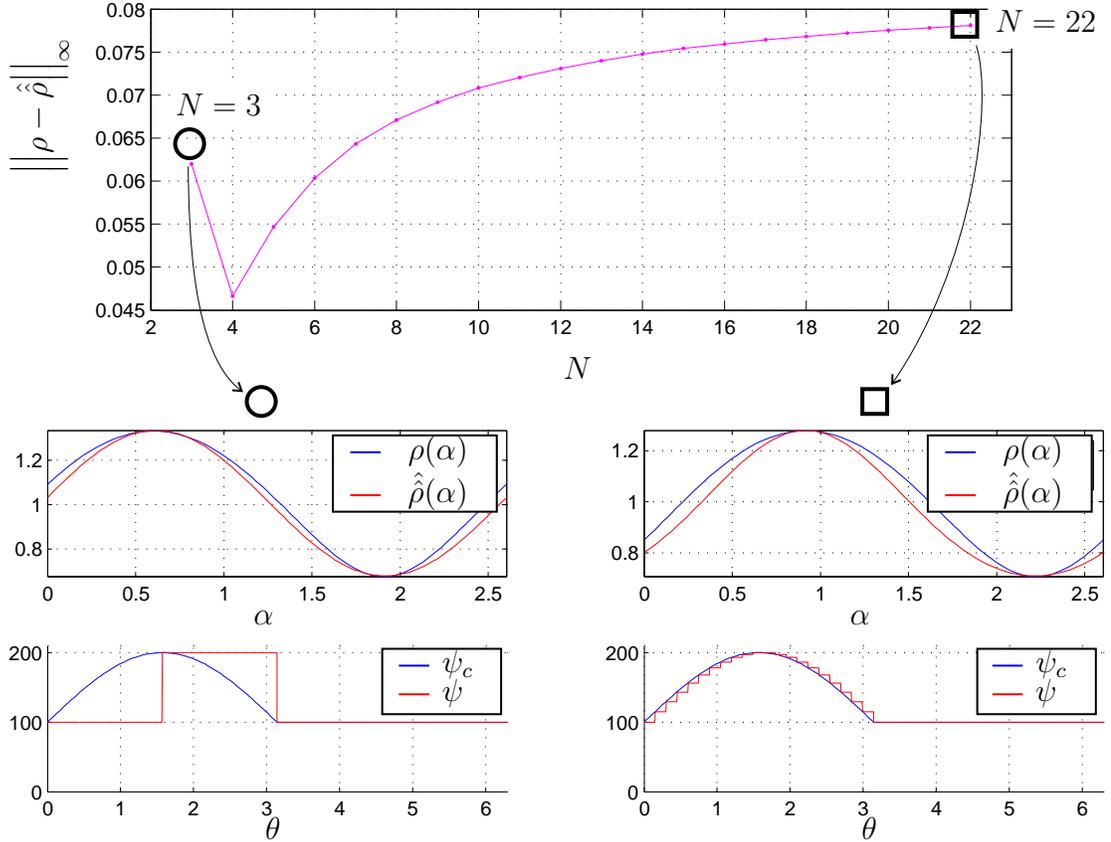


Figure B.11: The limit approximation error in the basic forcing function,  $\hat{\rho}(\alpha)$ , as the number of cells in the shape function,  $N$ , increases. The continuous shape template is a half period of a sinusoid function. The small action approximate is obtained by uniform sampling-and-hold.

instead of the original ones. Although the stability of the target behavior is not guaranteed, our numerical studies suggest that this approximate design procedure can successfully handle a wide spectrum of target behaviors.

An important byproduct of our discussion will be the identification of several interesting relationships between the generic shape features and the stability properties of the coupled system. Our numerical studies suggest that these relationships persist in the original setting as well as in a wider family of clock driven setups like those

discussed in Chapter C.

In this discussion we will limit our attention to a special subset of small action shape functions,  $\psi(\theta)$ , where the lift-off and last cell stiffness values are identical,  $\xi_e = \xi_N$ . Note that under this condition the approximate loss term,  $\hat{\epsilon}(\alpha)$ , is a constant,

$$\hat{\epsilon}(\alpha) = \zeta \exp \left[ -\mu \left[ \beta_{[1,e-1]} + \frac{1}{\gamma_N} \left( \pi - \sum_{k=1}^{e-1} \gamma_k \beta_k \right) \right] \right], \quad (\text{B.35})$$

and therefore, the variations of the basic forcing function,  $\hat{\rho}(\alpha)$ , governed by the basic cumulative action term,  $\hat{L}(\alpha)$ . Section B.6.1 will introduce a vector representation for the individual approximate actions,  $\hat{l}_i(\alpha)$ , as well as the basic cumulative action term,  $\hat{L}(\alpha)$ . We will point out several advantages of this representation in Section B.6.2 which will be employed in Section B.6.3 to demonstrate how to design a small action shape of a certain kind.

### B.6.1 Action Vectors

The approximate action term,  $\hat{l}_i(\alpha)$ , in (B.25), can be reorganized as

$$\hat{l}_i(\alpha) = \sqrt{(1 + A_i) \left( 1 + \frac{A_i}{1 - A_i} \cos(2\gamma_N \alpha + 2S_i) \right)}.$$

We will represent the phase shifted and scaled sinusoid term of the approximate action expression,  $\hat{l}_i(\alpha)$ , by an “action vector,”

$$\mathbf{l}_i := \frac{A_i}{1 - A_i} \begin{bmatrix} \cos(2S_i) \\ \sin(2S_i) \end{bmatrix}, \quad (\text{B.36})$$

defined in the 2-dimensional “action space,”  $\mathcal{L} := \mathbb{R}^2$  whose bases are  $\cos(2\gamma_N \alpha)$  and  $-\sin(2\gamma_N \alpha)$ . An alternative expression for the the approximate action term is given by

$$\hat{l}_i(\alpha) = \sqrt{(1 - A_i) [1 + |\mathbf{l}_i| \cos(2\gamma_N \alpha + \angle \mathbf{l}_i)]} \quad (\text{B.37})$$

where  $|\mathbf{l}_i| \in \overline{\mathbb{R}^+}$  is the magnitude of the action vector and  $\angle \mathbf{l}_i \in S^1 := (0, 2\pi) / \{0, 2\pi\}$  is the phase of the action vector — the angle between the positive cos-axis and the action vector,  $\mathbf{l}_i$ , in the counter clock-wise direction. Figure B.12 depicts typical action vectors for a relaxation and a stiffening action. Employing this alternative expression in (B.37) the basic cumulative action term,  $\hat{L}(\alpha)$ , can be rewritten as

$$\hat{L}(\alpha) := \sqrt{\left[ \prod_{i=1}^e (1 - A_i) \right] \left[ 1 + |\mathbf{l}_c| \cos(2\gamma_N \alpha + \angle \mathbf{l}_c) \right]}. \quad (\text{B.38})$$

where

$$\mathbf{l}_c = \sum_{i=1}^e \mathbf{l}_i. \quad (\text{B.39})$$

is the vector sum of all action vectors,  $\mathbf{l}_i$ , which we will refer as the “cumulative action vector.” Figure B.13 illustrates the cumulative action vector,  $\mathbf{l}_c$ , for a simple two action mode sequence,  $\mathcal{GAG}(2, N)$ .

## B.6.2 A Visual Design Tool

Recall that for the particular set of small shape configurations, where we choose the lift-off and last cell stiffness values to be identical,  $\xi_e = \xi_N$ , the approximate loss term,  $\hat{\epsilon}(\alpha)$ , is a constant. Therefore, the extrema of the basic forcing function,  $\hat{\rho}(\alpha)$ , are governed by the basic cumulative action term,  $\hat{L}(\alpha)$ . Hence, the action vector representation provides us a visual tool to study the basic forcing function as a function of the shape configuration,  $\mathbf{o}$ , and also relate it to the approximate invariant delay space,  $\hat{\mathcal{D}}^*$ .

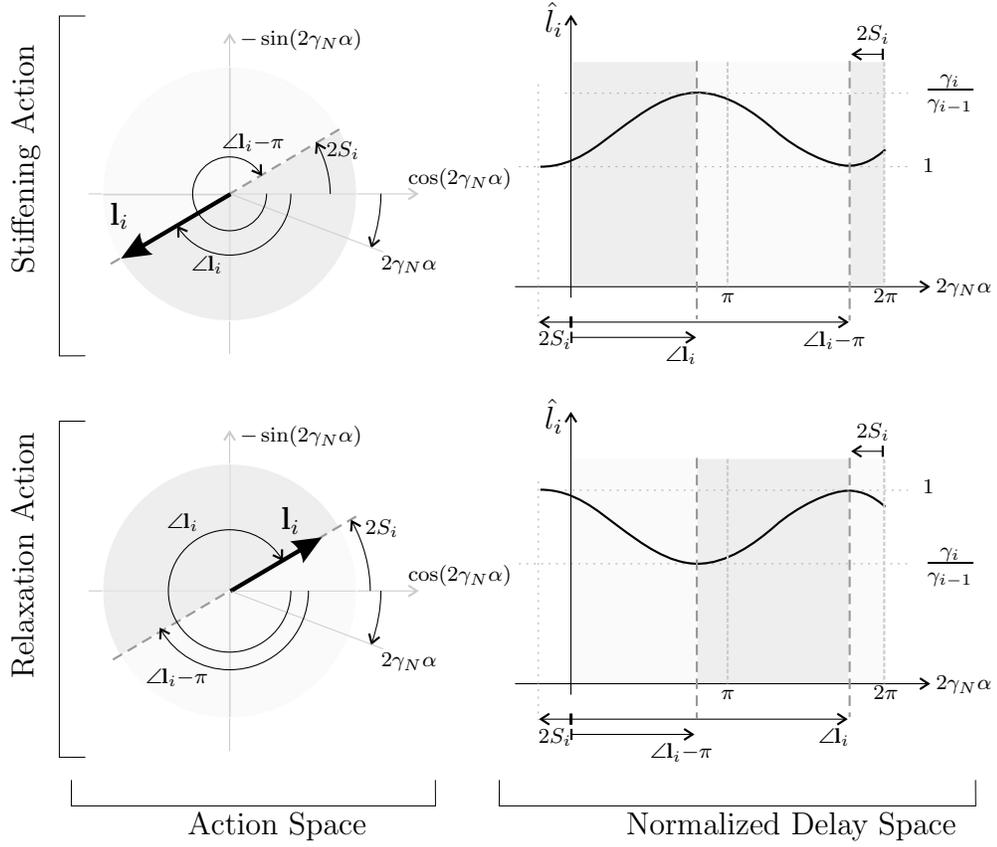


Figure B.12: Vector representations of basic actions,  $\hat{l}_i$ , for stiffening (top) and relaxation (bottom) actions. On the left column the action vectors,  $\mathbf{l}_i$ , are depicted in the action space,  $\mathcal{L}$ . The sketches on the right are the corresponding basic action terms,  $\hat{l}_i(\hat{\alpha})$ , plotted against the normalized delay,  $\hat{\alpha} := 2\gamma_N\alpha$ . The phase of the action vector,  $\Delta_i$ , indicates the delay where the associated basic action term,  $\hat{l}_i(\alpha)$ , has its local extrema points. The action space,  $\mathcal{L}$ , is partitioned into two cells: 1) a clock-wise half (light gray); and 2) a counter clock-wise half (dark gray). The basic action term,  $\hat{l}_i(\alpha)$ , is monotonically decreasing in the former and monotonically increasing in the latter.

By definition, the basic cumulative action term,  $\hat{L}(\alpha)$ , is a  $\pi/\gamma_N$  periodic function whose range is characterized by its mean,

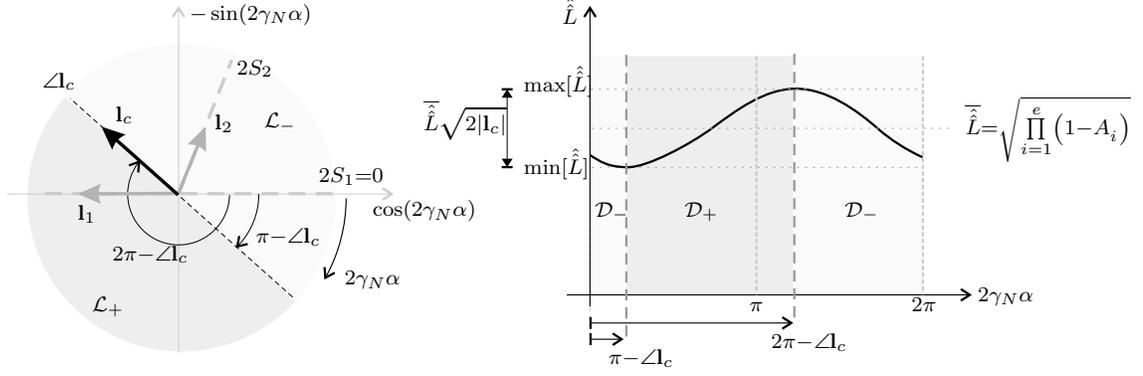


Figure B.13: An illustration of the action vectors for a two action fundamental repeatable mode sequence,  $\mathcal{GAG}(2, N)$ , as those investigated in [1]. The first action is a stiffening action,  $l_1$ , which is followed by a relaxation action,  $l_2$ . By definition, the first action vector,  $\mathbf{l}_1$ , is aligned with the  $\cos$ -axis,  $S_1 = 0$ . The cumulative action vector,  $\mathbf{l}_c$ , (black arrow) is the vector sum of the individual action vectors,  $\mathbf{l}_i$  (gray arrows). The phase of the cumulative action vector,  $\angle\mathbf{l}_c$ , determines the delay states where the basic cumulative action term,  $\hat{L}(\alpha)$ , has local extrema. The range of the basic cumulative action term is a function of the magnitude of the cumulative action vector,  $|\mathbf{l}_c|$ . Those value at the end of each ray indicates the angle between the ray and the positive  $\cos$ -axis in the counter clock-wise direction. All other angles are explicitly indicated.

$$\bar{\hat{L}} = \sqrt{\prod_{i=1}^e (1 - A_i)}, \quad (\text{B.40})$$

and its the peak-to-peak variation magnitude,

$$\max[\hat{L}] - \min[\hat{L}] = \bar{\hat{L}}\sqrt{2|\mathbf{l}_c|}, \quad (\text{B.41})$$

where  $|\mathbf{l}_c|$  is the magnitude of the cumulative action vector. The basic cumulative action term,  $\hat{L}(\alpha)$ , has no inflection points — points where both first and second

derivative of the function vanishes — but only two extrema over the delay space: a single minimum at  $\alpha_{min} \in \mathcal{D}$ ; and a single maximum at  $\alpha_{max} \in \mathcal{D}$ . Hence, the sign of the first derivative of the basic cumulative action term defines a partition over the delay space,  $\mathcal{D} = \mathcal{D}_+ \cup \mathcal{D}_- \cup \{\alpha_{min}, \alpha_{max}\}$ , such that the basic cumulative action term is strictly monotonically increasing over  $\mathcal{D}_+$ , and strictly monotonically decreasing over  $\mathcal{D}_-$ . Recall that basic forcing function,  $\hat{\rho}(\alpha)$ , is a scaled version of the basic cumulative action term,  $\hat{L}(\alpha)$ , and therefore, shares the same monotonicity.

The phase state,  $\angle$ , of the action space,  $\mathcal{L}$ , is related to the delay state,  $\alpha$ , of the Poincaré space,  $\mathcal{P}$ , by an affine map,  $u : \mathcal{D} \rightarrow S^1$ ,

$$\angle = u(\alpha) := 2\pi - 2\gamma_N \alpha. \quad (\text{B.42})$$

It directly follows the definition in (B.38) that the phase of the cumulative action vector,  $\angle \mathbf{l}_c$ , identifies the delay values where the basic cumulative action term,  $\hat{L}(\alpha)$ , has its extrema points,

$$\alpha_{min} = \arg \max_{\alpha \in \mathcal{D}} [\hat{L}(\alpha)] = \arg \max_{\alpha \in \mathcal{D}} [\hat{\rho}(\alpha)] = \frac{2\pi - \angle \mathbf{l}_c}{2\gamma_N}, \quad (\text{B.43})$$

$$\alpha_{max} = \arg \min_{\alpha \in \mathcal{D}} [\hat{L}(\alpha)] = \arg \min_{\alpha \in \mathcal{D}} [\hat{\rho}(\alpha)] = \frac{\pi - \angle \mathbf{l}_c}{2\gamma_N}.$$

Furthermore, the cumulative action vector,  $\mathbf{l}_c$ , effectively partitions the action space,  $\mathcal{L}$ , into two halves that are related to the delay space partition: 1) a clock-wise half,  $\mathcal{L}_- := u(\mathcal{D}_-)$ ,

$$\mathcal{L}_- := \{\mathcal{L} \mid \angle \in (\angle \mathbf{l}_c - \pi, \angle \mathbf{l}_c)\};$$

and 2) a counter clock-wise half,  $\mathcal{L}_+ := u(\mathcal{D}_+)$ ,

$$\mathcal{L}_+ := \{\mathcal{L} \mid \angle \in (\angle \mathbf{l}_c, \angle \mathbf{l}_c + \pi)\}.$$

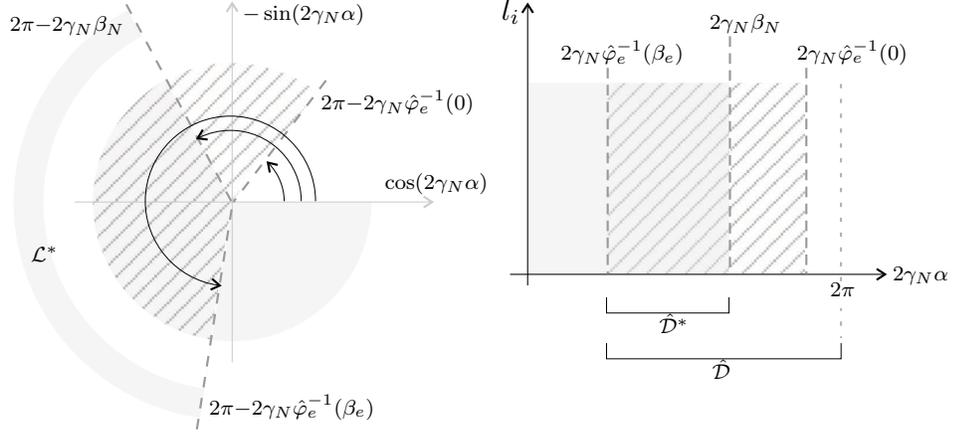


Figure B.14: The approximate invariant delay interval. Those numbers at the end of each ray indicate the angle between the ray and the positive  $\cos$ -axis in the counter clock-wise direction.

It directly follows from the definition in (2.34), that the approximate invariant delay interval,  $\hat{\mathcal{D}}^*$ , corresponds to a phase interval in the action space,  $\mathcal{L}^* := u(\hat{\mathcal{D}}^*)$ , between a lower bound,

$$\partial\mathcal{L}^*_- := 2\pi - 2\gamma_N \max\{0, \hat{\varphi}_e^{-1}(\beta_e)\}, \quad (\text{B.44})$$

and an upper bound,

$$\partial\mathcal{L}^*_+ := 2\pi - 2\gamma_N \min\{\beta_N, \hat{\varphi}_e^{-1}(0)\}. \quad (\text{B.45})$$

as depicted in Figure B.14.

### B.6.3 Designing Spike Shape Functions

#### Spike Shape Function

We limit our attention to a family of “spike shape functions,”  $\psi(\theta)$ , whose generic form is illustrated in Figure B.15. We divide the spike shape function,  $\psi(\theta)$ , into

three parts: 1) the spike cells,  $\Theta_1 \cup \dots \cup \Theta_{e-1}$ ; 2) a lift-off cell,  $\Theta_e$ ; and 3) a nominal cell,  $\Theta_N$ .

We will presume that the coupled system will operate in a fundamental repeatable mode sequence,  $\mathcal{GAG}(e, N)$ , where the entire spike portion coincides with the stance mode. The mechanical hopper lifts off in the lift-off cell,  $\Theta_e$ , and touches down in the last cell,  $\Theta_N$ . A typical target mode string is depicted along side the sketch of the spike shape function in Figure B.15.

The spike portion of the shape function, whose overall duration will be denoted by  $\beta_{[1,e-1]} = \bar{\beta}$ , consists of  $(2L - 1)$  cells of identical duration,  $\beta_i = \bar{\beta}/(2L - 1)$  for  $i = 1, 2, \dots, e - 1$ . Within the spike portion the shape function,  $\psi(\theta)$ , takes  $L$  different values. In our discussions,  $L$  will be called the “number of levels” of the spike shape. Without any loss of generality we will assume that  $L = e - 1$ . Consecutive levels of the spike,  $\xi_i$  and  $\xi_{i+1}$ , are related such that the ratio of the corresponding natural oscillation frequencies is a constant,  $\gamma_{i+1}/\gamma_i = c$ , where  $c$  will be called the “spike ratio.” Cells of the spike portion are separated by  $2L$  actions: first  $L$  actions are of stiffening type; and the last  $L$  actions are of relaxation type. Note that stiffening and relaxation actions are paired such that for every stiffening action that changes the stiffness from  $\xi_i$  to  $\xi_{i+1}$  there is a relaxation action that changes the stiffness from  $\xi_{i+1}$  to  $\xi_i$ . We will call these actions “reciprocal actions” of the spike.

Hence, a spike shape function,  $\psi(\theta)$ , is specified by five parameters: 1) spike duration,  $\bar{\beta}$ ; 2) number of levels,  $L$ ; 3) the spike ratio,  $c$ ; 4) duration of the lift-off cell,  $\beta_e$ ; and 5) the duration of the last cell,  $\beta_N$ .

### Shape Design Algorithm

The task specification in Section B.1 effectively defines the desired fixed point speed,  $\nu^*$ , according to (B.1). The computational algorithm, whose flow diagram can

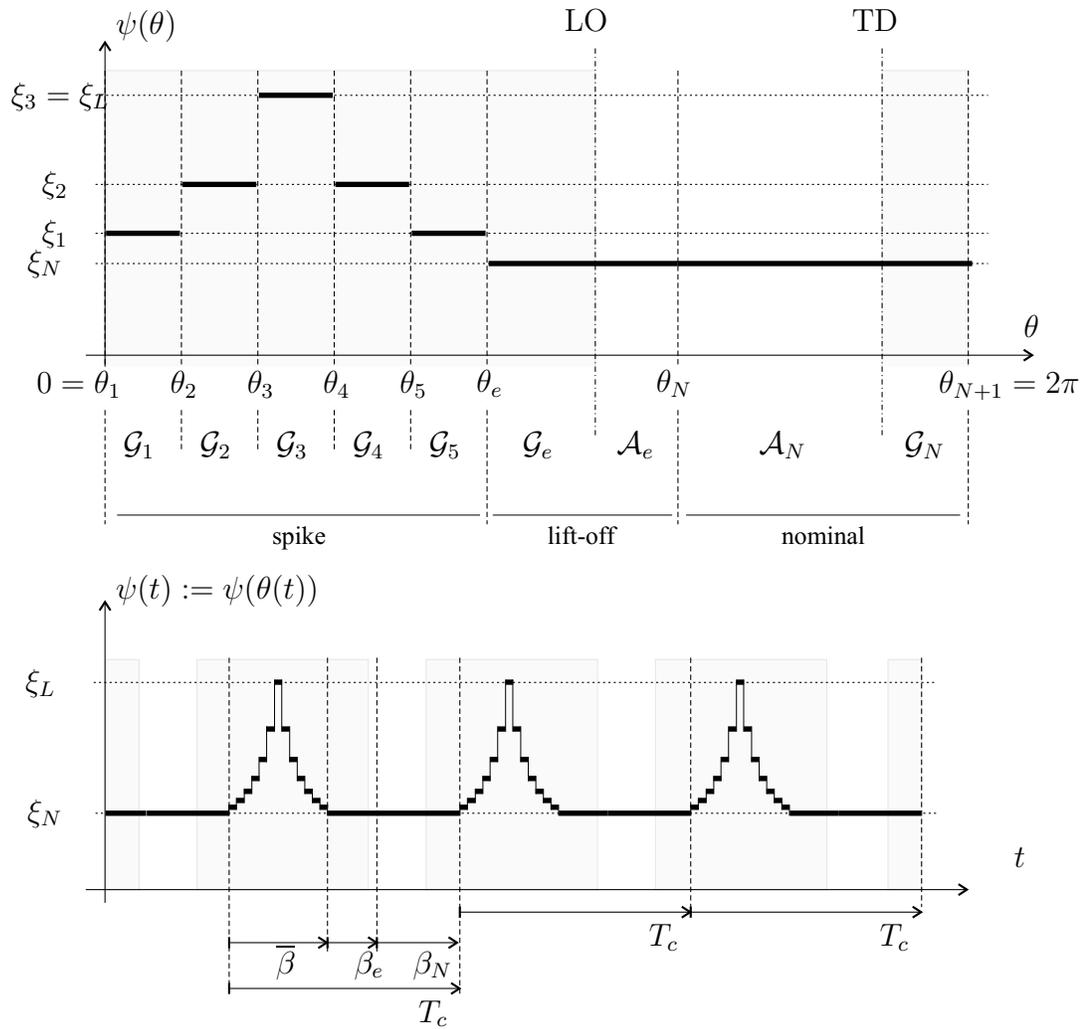


Figure B.15: An illustration of the spike shape function. The profile is partitioned into three cells: 1) spike portion; 2) lift-off cell; and 3) nominal cell. The spike portion is composed of  $2L - 1$  cells of identical duration where  $L$  is the number of levels in output spans in spike portion. The target operating regime is a fundamental repeatable mode sequence. The mechanical system transitions into the aerial mode in the lift-off cell,  $\Theta_e$ . The ground contact occurs in the nominal cell,  $\Theta_N$ . Notice that  $N = e + 1$ .

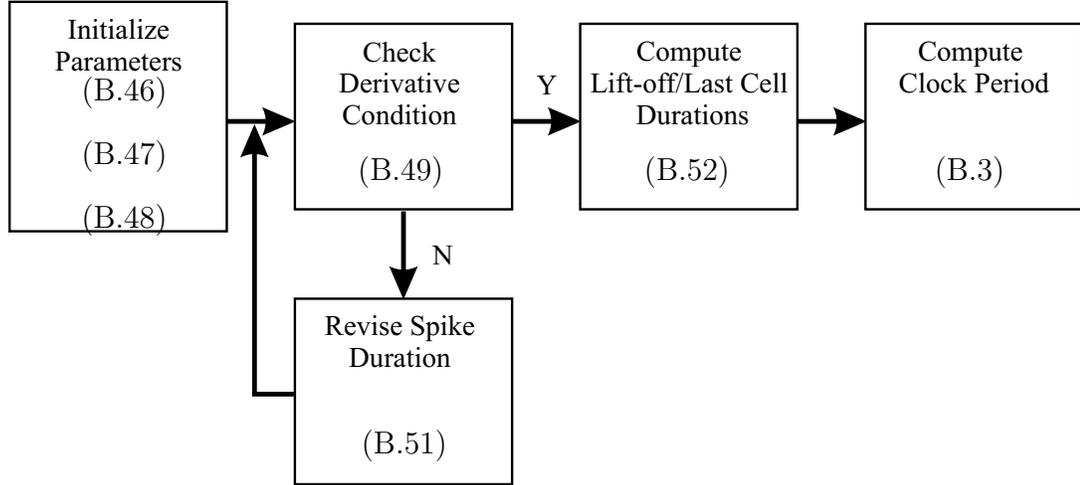


Figure B.16: The flow chart of the (approximate) spike shape function,  $\psi(\theta)$ , design procedure. Each step refers to those equations that govern the computational process to be performed.

be found in Figure B.16, consists of three parts: 1) initialization; 2) correction of the spike duration,  $\bar{\beta}$ ; and 3) the computation of the clock period,  $T_c$ .

The initialization is where several parameters of the spike shape are chosen to start the process off. Two parameters of the spike shape function,  $\psi(\theta)$ , namely, the nominal stiffness,  $\xi_n$ ; and the number of levels,  $L$ , are considered to be free parameters that we choose such that

$$L > 2, \quad \xi_e = \xi_n, \quad \xi_N = \xi_n \quad (\text{B.46})$$

where the nominal stiffness,  $\xi_n$ , which constitutes the smallest stiffness, must be chosen in accordance with the lower bound defined in (B.6). Recall that the accuracy of the approximate expressions, which directly affects the accuracy and validity of this approximate design approach, improves with higher nominal stiffness,  $\xi_n$ , and larger number of levels,  $L$ . Therefore, larger values for both free parameters are desirable.

The initial spike duration,  $\bar{\beta}_0$ , is chosen to be half the stance duration with constant nominal stiffness,

$$\bar{\beta}_0 := \frac{\pi}{2\gamma_N}. \quad (\text{B.47})$$

Recall that since the lift-off and last cell stiffnesses are chosen to be identical,  $\xi_e = \xi_N$ , the approximate loss term,  $\hat{\epsilon}(\alpha)$ , is a constant for spike shape which allows us to reorganize the basic forcing function,  $\hat{\rho}(\alpha) = \bar{\hat{\rho}}\sqrt{1 + |\mathbf{l}_c| \cos(2\gamma_N\alpha + \angle\mathbf{l}_c)}$ , where

$$\bar{\hat{\rho}} := \zeta \exp\left[-\frac{\mu\pi}{\gamma_N}\right] \exp\left[-\mu\bar{\beta}\left(1 - \frac{c^{(L+1)/2} + 2\sum_{k=1}^L c^k}{L}\right)\right] \left[\frac{1}{4}(1+c^2)\left(1 + \frac{1}{c^2}\right)\right]^{\frac{L+1}{4}}.$$

is the mean of the basic forcing function,  $\hat{\rho}(\alpha)$ . We compute the spike ratio,  $c$ , such that the mean of the basic forcing function,  $\bar{\hat{\rho}}$ , equal to unity for the initial spike duration,  $\bar{\beta}_0$ , as given by

$$\bar{\hat{\rho}}(c, \bar{\beta}_0) = 1. \quad (\text{B.48})$$

Since the mean of the basic forcing function,  $\bar{\hat{\rho}}$ , is monotonic in the spike ratio,  $c$ , there is a unique solution to (B.48) which can be obtained by simple root-finding algorithms. Moreover, for physically relevant lossy setups,  $\mu > 0$  and  $\zeta \in (0, 1)$ , the spike ratio is guaranteed to be larger than unity,  $c > 1$ , which defines a spike shape function,  $\psi(\theta)$ , that is bounded by the nominal stiffness from below,  $\psi(\theta) \geq \xi_n$ , as intended.

At the parameters selected by the initialization step the basic forcing function,  $\hat{\rho}(\alpha)$ , intersect with the unity at a delay,  $\alpha^*$ , where has its largest derivative. However, this shape configuration may fail the derivative condition which must be satisfied so that the clock period falls into the stable period interval,  $\mathcal{T}_s$ . The second step of the

design process, which takes an iterative form, corrects the spike shape configuration,  $\mathbf{o}$ , so that the bounded derivative condition in (B.7) is also satisfied. Since we will only consider those unity intersections of the basic forcing function,  $\hat{\rho}(\alpha)$ , where the basic forcing function is monotonically increasing where the lower bound is satisfied. Substitution of the surrogate expressions simplifies the upper derivative bound in (B.7) to

$$\hat{\rho}'(\alpha^*) < \frac{1}{\nu^*} \quad (\text{B.49})$$

The first derivative of the basic forcing function is given by

$$\hat{\rho}'(\alpha) = \bar{\hat{\rho}} \gamma_N |\mathbf{l}_c| \frac{\sin(2\gamma_N \alpha + \angle \mathbf{l}_c)}{\sqrt{1 + |\mathbf{l}_c| \cos(2\gamma_N \alpha + \angle \mathbf{l}_c)}}$$

whose maximum value,

$$\max \left[ \hat{\rho}'(\alpha) \right] = \bar{\hat{\rho}} \gamma_N |\mathbf{l}_c|, \quad (\text{B.50})$$

is a monotonic function of the cumulative action magnitude,  $|\mathbf{l}_c|$ , and the basic forcing function mean,  $\bar{\hat{\rho}}$  — both of which are monotonically decreasing functions of the spike duration,  $\bar{\beta}$ . Note that decreasing the spike duration,  $\bar{\beta}$ , not only decreases the derivative of the basic forcing function,  $\hat{\rho}'(\alpha)$ , at all delay values,  $\alpha \in \mathcal{D}$ , but also shifts the basic forcing function,  $\hat{\rho}(\alpha)$ , down (since the mean,  $\bar{\hat{\rho}}$ , is decreasing) resulting in a unity crossing at a delay,  $\alpha^*$ , where the derivative of the basic forcing function,  $\hat{\rho}'(\alpha^*)$ , is less than its maximum in (B.50). Hence, the derivative of the basic forcing function at unity crossing,  $\hat{\rho}'(\alpha^*)$ , is a monotonically decreasing function of the spike duration,  $\bar{\beta}$ , which we choose to employ in an iterative process,

$$\bar{\beta}_k = \bar{\beta}_{k-1} - \Delta \bar{\beta}, \quad (\text{B.51})$$

to correct the spike shape configuration,  $\mathbf{o}$ , in order to satisfy (B.49) if it was not met at the end of the initialization.

The final step in shape design is the computation of the lift-off and last cell durations. Note that the basic forcing function,  $\hat{\rho}(\alpha)$ , intersects with the unity in the interval specified by the phase of the cumulative action vector,

$$\alpha^* \in \left( \frac{\pi/2 - \angle \mathbf{1}_c}{2\gamma_N}, \frac{\pi - \angle \mathbf{1}_c}{2\gamma_N} \right),$$

which we choose to include in the approximate invariant delay interval,  $\hat{\mathcal{D}}^*$ , by setting the lift-off and last cell durations as follows

$$\beta_e = \hat{\varphi}_e^{-1} \left( \frac{\pi/2 - \angle \mathbf{1}_c}{2\gamma_N} \right), \quad \beta_N = \frac{\pi - \angle \mathbf{1}_c}{2\gamma_N}. \quad (\text{B.52})$$

We conclude the design process by the computation of the proper clock period,  $T_c$ , according to (B.3).

## APPENDIX C

# Clock Driven 1-DOF Generalized Mechanical Systems

The 1-DOF linear prismatic hopper (LPH) in Section 2.1.1 offers an illustrative mechanical model. Its relatively simple dynamics capture the basic hybrid nature of a typical legged locomotion behavior consisting of alternating (partially) controllable stance phases and uncontrollable aerial phases. While this abstract setup allows us to derive analytical results, it fails to capture many physically relevant details that are bound to appear in physical implementations. This appendix generalizes the basic ideas, presented in Chapter 2, into more physically relevant mechanical settings.

### C.1 Leg Kinematics

The simplicity of the model in Section 2.1.1, which paved the way for the formal analysis in Section 2.4, can be partially attributed to the prismatic leg morphology which offers a trivial relationship between the actuator force and the ground reaction force. However, in order to satisfy a number of constraints and overcome several manufacturing problems physical robotic platform often adopt more complicated mor-

phologies consisting of a combination of rotational and prismatic joints. Surely, the kinematic properties of legs strongly influence the forces that operate on the center of mass (COM).

This section will concentrate on the effects of leg kinematics in the context of clock driven mechanisms. For this discussion Section C.1.1 introduces the vertical watch spring model (VWSM) — a modified 1-DOF hopper model. We will demonstrate that using feedback the closed-loop VWSM dynamics can be rendered identical to that of the linear prismatic hopper (LPH) where the results of Chapter 2 and Chapter 3 directly apply. According to the terminology of [99] in this closed-loop setting the linear prismatic hopper (LPH) dynamics is “anchored” into the VWSM. Of course, anchoring, a moral equivalent of inverse dynamics control, requires considerable sensory information. In Section C.1.3 we will look into the clock driven VWSM that does not employ any feedback. We will present numerical results suggesting that local stability can be achieved in the absence of feedback. This case study will lead to a more general discussion in Section C.2.

### C.1.1 Vertical Watch Spring Model

The vertical watch spring model (VWSM), illustrated in Figure C.1, is a 1-DOF Lagrangian mechanical system. It consists of a point body mass,  $m$ , and a rigid massless leg of length  $l$ . The body is holonomically constrained to move along the vertical and its height is denoted by  $x_1$ . The leg is attached to the body at one of its ends through a rotational joint that we will refer as the “hip.” The other distal end of the leg is called the “toe.” During ground contact the toe slides along the ground without any friction. The angle between the vertical and the leg in the counter clockwise direction is the hip angle and denoted by  $\phi$ . The hip actuator takes the form of an ideal torque generator whose output torque is denoted by  $\tau$ .

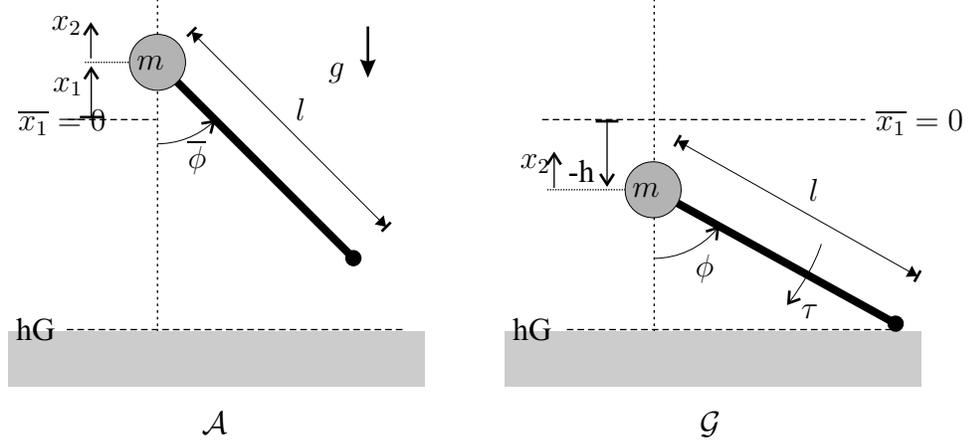


Figure C.1: 1-DOF Vertical Watch Spring Model (VWSM) in ground contact,  $\mathcal{G}$ , (right); and in flight,  $\mathcal{A}$ , (left).

In order to ease the comparison between the LPH and VWSM we choose to study the VWSM in the same dimensionless height coordinates introduced in the LPH analysis in Section 2.1.1. We let the set of physically valid body height,  $x_1 \in (-1, \infty)$ , be those configurations where the body is above the ground surface which is located at  $x_1 = -1$  in this dimensionless setting. Accordingly, we define the physical coordinate system for the VWSM,  $\mathcal{X} := [(-1, \infty) \times \mathbb{R}] - \{0\}$ . We exclude the origin where the trivial solution resides since this analysis only concerns cyclic limit behaviors.

The nonlinear *hybrid* dynamics of VWSM,

$$\dot{\mathbf{x}} = \tilde{\mathbf{f}}(\mathbf{x}, \xi), \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathcal{X}, \quad (\text{C.1})$$

has two modes: a stance mode,  $\mathcal{G}$ , illustrated in Figure C.1(right); and an aerial mode,  $\mathcal{A}$ , shown in Figure C.1(left). Correspondingly, the state space of the VWSM,  $\mathcal{X}$ , is partitioned into two charts: a stance set,  $\mathcal{X}_{\mathcal{G}} := \{\mathbf{x} \in \mathcal{X} | x_1 < 0\}$ , where the dynamics is governed by

$$\dot{\mathbf{x}} = \tilde{\mathbf{f}}_{\mathcal{G}}(\mathbf{x}) := \begin{bmatrix} x_2 \\ \left[ \frac{l^2 - x_1^2}{l^2} \right] \tau \end{bmatrix} \quad \mathbf{x} \in \mathcal{X}_{\mathcal{G}}; \quad (\text{C.2})$$

and, an aerial set,  $\mathcal{X}_{\mathcal{A}} := \{\mathbf{x} \in \mathcal{X} | x_1 > 0\}$ , where the body moves under the influence of unit gravitational acceleration,

$$\dot{\mathbf{x}} = \tilde{\mathbf{f}}_{\mathcal{A}}(\mathbf{x}) := \begin{bmatrix} x_2 \\ -1 \end{bmatrix} \quad \mathbf{x} \in \mathcal{X}_{\mathcal{A}}. \quad (\text{C.3})$$

In order to simplify the transition boundaries that govern the mode changes we will assume that the hip torque,  $\tau$ , is zero for all hip angles smaller than a rest angle,  $\phi_0 := \sigma(0)$ , and positive otherwise. Hence, during stance mode,  $\mathcal{G}$ , as the leg supports the body the hip angle,  $\phi$ , and the body height,  $x_1$ , are kinematically related according to

$$\phi := \sigma(x_1) = \arctan \left[ \frac{\sqrt{l^2 - x_1^2}}{x_1} \right], \quad (\text{C.4})$$

while in the aerial mode,  $\mathcal{A}$ , the (massless) leg is hold at the rest angle,  $\phi_0$ . This setup leads to a touchdown boundary,  $\partial\mathcal{X}_{\mathcal{G}}^- := \{\mathbf{x} \in \mathcal{X} | x_1 = 0 \wedge x_2 < 0\}$ , and a lift-off boundary,  $\partial\mathcal{X}_{\mathcal{G}}^+ := \{\mathbf{x} \in \mathcal{X} | x_1 = 0 \wedge x_2 > 0\}$ , that are both at the same body height,  $x_1 = 0$ .

### C.1.2 Anchoring LHP Dynamics

Despite its different mechanical configuration the VWSM is equivalent to the LHP model. In fact, one can embed the LHP dynamics into the VWSM using a carefully tailored actuation mechanism in the hip as a function of proprioceptive sensing,

$$\tau := \frac{ml^2}{\sqrt{l^2 - [\sigma^{-1}(\phi)]^2}} \left[ -\xi^2 \sigma^{-1}(\phi) - 2\mu D\sigma^{-1}(\phi) \dot{\phi} \right]. \quad (\text{C.5})$$

This leads to effective VWSM vertical dynamics that are identical to that of the LPH in Section 2.1.1.

It immediately follows that the feed-forward control of this anchored LHP dynamics according to the control mechanism in Section 2.1.2 yields locally asymptotically stable hopping behavior.

### C.1.3 VWSM with Hook’s Law Hip Spring

Embedding a target dynamics as in Section C.1.2 requires continuous high bandwidth sensory feedback and a considerable computational power. Although our case study presents a simple framework in more realistic settings the underactuated nature of the physical system and other constraints may make the embedding inaccurate. Furthermore, model based controllers are prone to modeling errors which are bound to occur.

On the other hand, the very special form of the LHP dynamics is not mandatory to achieve dynamically stable behaviors with open-loop controllers.

To demonstrate this we will consider another version of the VWSM where the hip actuator is replaced by a lossy torsional Hooke’s law spring with adjustable stiffness. By choosing the rest configuration of the torsional spring to coincide with the rest position,  $\phi_0$ , we satisfy the torque assumption and maintain the transition boundaries unchanged. The aerial dynamics of the VWSM remains to be identical to (C.3). On the other hand, the stance dynamics takes a much more complicated form,

$$\dot{\mathbf{x}} = \tilde{\mathbf{f}}_{\mathcal{G}}(\mathbf{x}) := \begin{bmatrix} x_2 \\ \left[ \frac{l^2 - x_1^2}{l^2} \right] [\xi^2 \sigma(x_1) - 2\mu D \sigma(x_1) x_2] \end{bmatrix} \quad \mathbf{x} \in \mathcal{X}_{\mathcal{G}}; \quad (\text{C.6})$$

The ground dynamics is too complicated to derive analytic results as we did in Chapter 2. Yet, we will present numerical simulation that exhibit locally asymptoti-

cally stable hopping behavior.

## C.2 Clock Driven 1-DOF Cyclic Systems

The detailed analysis in Chapter 2 and the associated design discussion in Chapter B are both founded on a very specific mechanical hopper model, namely, the leg is modeled as a linear prismatic spring with adjustable leg stiffness. Although this model is a useful tool to investigate the basics of the open-loop control it is far from being an accurate representation of physical mechanisms. Physical springs often present non-linear force laws. Furthermore, as we discussed in Section C.1, the kinematic properties of the leg have significant authority over the dynamical properties of the mechanical system.

In Chapter 2 we identified several sufficient conditions for instability and stability of the return map's fixed points. Now, we will present a preliminary numerical study on a wider family of clock driven mechanical hoppers. Our numerical studies suggest that the stability and instability properties persist in this family. In addition, we will address the issue of how to pick the spring properties. Our result suggest that the stability properties of the coupled system vary according to the spring properties.

We will first introduce a generalized family of cyclic mechanical systems in Section C.2.1. Note that 1-DOF hoppers fall into this family. Next, we will derive a Poincaré return map for the clock-driven mechanical system in Section C.2.3. We will outline a local stability analysis of this return map in Section C.2.4 where we will present several numerical experiments that suggest that the stability of the coupled system is determined by the properties of the spring law. We will conclude in Section C.2.5 with a discussion of 1-DOF hoppers with nonlinear leg spring. We will numerically demonstrate that certain spring potentials give rise to favorable coupled behavior.

### C.2.1 A Family of Clock Driven 1-DOF Mechanical Systems

In this discussion, we will consider a family of 1-DOF dissipative Lagrangian mechanical systems defined in generalized position,  $x_1 \in \mathbb{R}$ , and generalized speed,  $x_2 := \dot{x}_1 \in T\mathbb{R}_{x_1} = \mathbb{R}$ , coordinates,

$$\dot{\mathbf{x}} = \mathbf{f}_\xi(\mathbf{x}) = \begin{bmatrix} x_2 \\ F_\xi^s(x_1) + F_\mu^d(x_1, x_2) \end{bmatrix}, \quad \mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} \in \mathcal{X} := \mathbb{R}^2, \quad (\text{C.7})$$

where  $F_\xi^s(x_1)$  is the adjustable spring force law and  $F_\mu^d(x_1, x_2)$  is the dissipation term.

The control input to the mechanical system is the adjustable parameter of the force law,  $\xi$ , which we will refer as the “spring parameter.” The spring parameter,  $\xi$ , selects the force profile from a one-parameter family,

$$F^s \in C^0[\mathcal{K} \times \mathcal{X}, \mathbb{R}], \quad (\text{C.8})$$

whose members satisfy a *non-negative stiffness* constraint,

$$\forall x_1 \in \mathcal{X}, \quad \text{and} \quad \xi \in \mathcal{K}, \quad -\frac{dF_\xi^s(x_1)}{dx_1} \geq 0,$$

as well as a *monotonic stiffening* property

$$\forall \xi_1, \xi_2 \in \mathcal{K}, \quad \xi_1 < \xi_2 \quad \Rightarrow \quad |F_{\xi_1}^s(x_1)| < |F_{\xi_2}^s(x_1)|.$$

We also assume that the force law,  $F_\xi^s(x_1)$ , is Lipschitz continuous with a fixed Lipschitz constant,  $L_{F^s}$ , over admissible configurations,  $\mathcal{X}$ .

Dissipation term,  $F_\mu^d(x_1, x_2)$ , is a function of both generalized position,  $x_1$ , and generalized speed,  $x_2$ . We will assume that the dissipation term,  $F_\mu^d(x_1, x_2)$ , is also globally Lipschitz continuous with a fixed global Lipschitz constant,  $L_{F^d}$ . A scalar

damping coefficient,  $\mu$ , continuously parameterizes the dissipation term,  $F_\mu^d(x_1, x_2)$ , in a strictly monotonic fashion,

$$\forall \mathbf{x} \in \mathcal{X}, \quad \text{and} \quad \mu \in \mathcal{B}, \quad \frac{\partial F_\mu^d(x_1, x_2)}{\partial \mu} > 0.$$

Without any loss of generality we will assume that the *lossless* case where  $F_\mu^d(x_1, x_2) \equiv 0$ , occurs when the damping coefficient vanishes,  $\mu = 0$ . It follows directly from the monotonicity of the dissipation term that for positive damping values,  $\mu > 0$ , the system is *lossy*,  $x_2 F_\mu^d(x_1, x_2) < 0$ , and for negative damping values,  $\mu < 0$ , the system is *gainy*,  $x_2 F_\mu^d(x_1, x_2) > 0$ .

The solution through the initial condition,  $\mathbf{x}(0) \in \mathcal{X}$ , generated by (C.7) operating at a constant parameter value,  $\xi$ , will be denoted by  $\mathbf{f}_\xi^t(\mathbf{x}(0))$ . Associated with (C.8) there are, of course, two important related function families: the potential,

$$U_\xi(x_1) := - \int_0^{x_1} F_\xi^s(s) ds;$$

and the total energy,

$$H_\xi(\mathbf{x}) := \frac{1}{2} x_2^2 + U_\xi(x_1).$$

It directly follows the definition of the force law,  $F_\xi^s(x_1)$ , that the potential,  $U_\xi(x_1)$ , is a convex function over the admissible set,  $\mathcal{X}$ , with a unique minima. We define *energy level set*,  $\mathcal{M}_\xi^h := \{\mathbf{x} \in \mathcal{X} | H_\xi(\mathbf{x}) = h\}$ , as the set of states with the same total energy,  $h$ , where the potential law,  $U_\xi(x_1)$ , is chosen by the spring parameter,  $\xi$ .

We construct the clock driven 1-DOF setup by coupling a clock controller of a kind introduced in Section 2.1.2 with the generic mechanical system in (C.7) such that the controller output,  $\psi(t)$ , modulates the spring parameter,  $\xi$ .

## C.2.2 Three Spring Types

In clock driven 1-DOF mechanical systems, there is no explicit mechanism that governs the synchronization of the phases of the mechanical system and the controller clock. Yet, in Chapter 2 we demonstrated that for a particular hopper model the clock and the mechanical system do synchronize under certain conditions. Our analysis revealed that the actions of the controller, modulation of spring stiffness, indirectly affect the mechanical phase speed through the relationship between the phase speed and the total mechanical energy. It directly follows that the stability properties of the coupled setup strongly depends on the relationship between the mechanical period,  $T_m$ , and the mechanical energy,  $h$ .

To facilitate the following discussions, we categorize the spring potentials into three groups:

$$\begin{aligned}
 &1) \text{ softening springs, } \frac{dT_m(h)}{dh} > 0; \\
 &2) \text{ linear springs, } \frac{dT_m(h)}{dh} = 0; \\
 &3) \text{ hardening springs, } \frac{dT_m(h)}{dh} < 0.
 \end{aligned} \tag{C.9}$$

The period-energy relationship strongly depends on the changes in stiffness as a function of the energy level. Consider a mechanical system of the form (C.7). In a lossless setting,  $\mu = 0$ , with constant spring parameter,  $\xi$ , the constant energy level sets, in the phase plane,  $\mathcal{X}$ , define closed curves,  $\mathcal{M}_\xi^h := \{\mathcal{X} \mid H_\xi(\mathbf{x}) = h\}$ . The period of mechanical oscillation,  $T_m$ , is a function of the total mechanical energy,

$$T_m(h) = \frac{d}{dh} \left[ \oint_{\mathcal{M}_\xi^h} x_2 dx_1 \right].$$

where the integral over the level set,  $\mathcal{M}_\xi^h$ , is the area enclosed by it. The sign of the first derivative of the mechanical period with respect to the total energy depends on the sign of the second derivative of the potential law, the stiffness. See Figure C.2

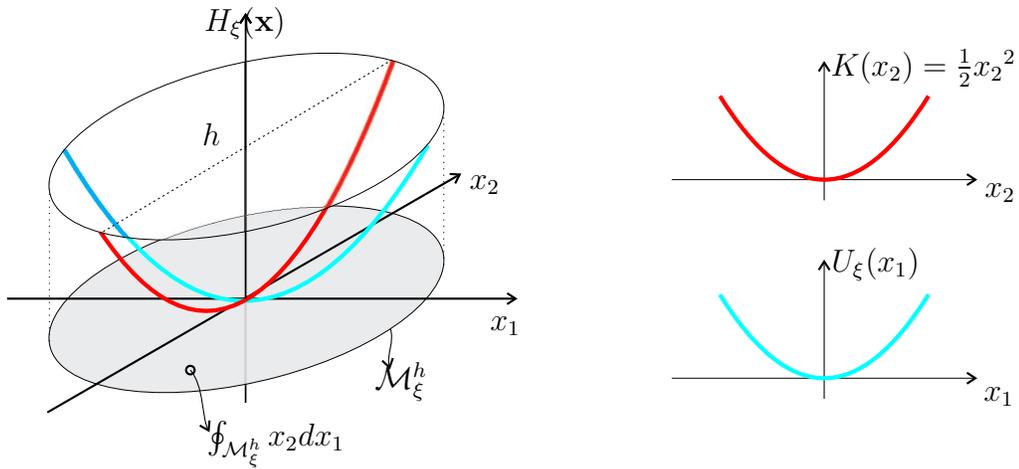


Figure C.2: The relationship between the area enclosed by the level set,  $\mathcal{M}_\xi^h$ , and the potential energy function,  $U_\xi(x_1)$ .

for a visual explanation. Hence, the spring classification in (C.9) effectively divides springs according to the relationship between stiffness and energy. Figure C.3 depicts three spring force profiles.

The rate at which the mechanical period,  $T_m$ , changes as a function of total mechanical energy,  $h$ , will be referred as the “softness measure.” Constrained in certain potential families the first derivative of the stiffness with respect to the energy can serve as a effective softness measure.

### C.2.3 Derivation of Generalized Return Map

For our discussion we will consider a clock driven (generic) 1-DOF mechanical system. The structure of the controller will be identical to that defined in Section 2.1.2. We limit our attention to those coupled operating regimes described by the fundamental repeatable mode sequences,  $\mathcal{GAG}(e, N)$ , introduced in Section 2.4. The basic structure of the analysis goes parallel to the study presented in Chapter 2. In fact, we will be using the same symbols to indicate these similarities and demonstrate

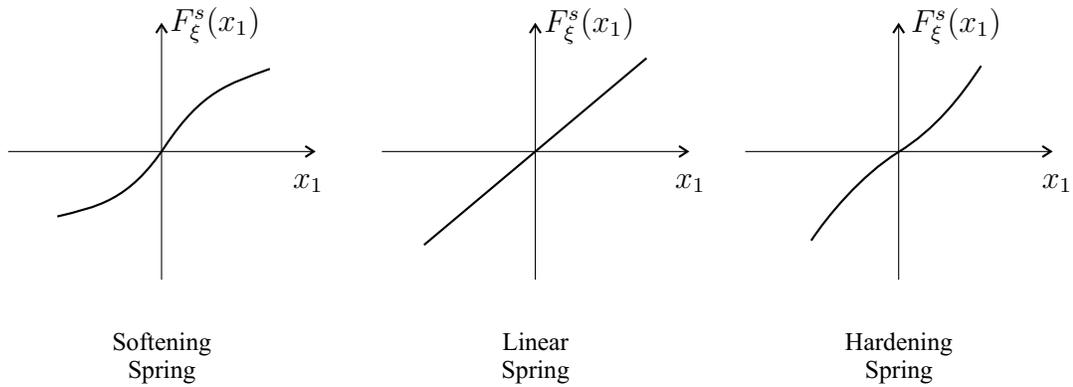


Figure C.3: Illustrations of the three spring types: softening (left); linear (middle); and hardening (right).

how the results of Chapter 2 inspire our work in this section.

The stability analysis of the coupled system will be done by Poincaré analysis as in Chapter 2. Sampling the flow of the coupled system at the event of intersection with the Poincaré section,

$$\Sigma := \{\mathcal{X} \times \Theta \mid x_1 = 0 \wedge x_2 < 0\},$$

relates the continuous-time flow of the coupled system to a discrete sequence. By the virtue of the mechanical dynamics in (C.7) the flow of the coupled system starting from  $\Sigma$  is guaranteed to return to it, and therefore,  $\Sigma$  is a good Poincaré section. We choose to map the independent variables of the Poincaré samples (the generalized speed,  $x_2$ , and the clock phase,  $\theta$ ) to an alternative Poincaré space,  $\mathcal{P}$  — defined by the transformation,  $\mathbf{n} : \mathcal{X} \rightarrow \mathcal{P}$ ,

$$\mathbf{p} = \begin{bmatrix} \nu \\ \alpha \end{bmatrix} = \mathbf{n}(\mathbf{x}, \theta) := \begin{bmatrix} H_{\xi_N}(\mathbf{x}) \\ 2\pi - \theta \end{bmatrix}.$$

Note that the physical interpretation of the Poincaré states are identical to those in Section 2.2.2. We define the “mechanical return” as the event where the mechanical

flow crosses from the fourth quadrant into the third quadrant. Namely, the delay state,  $\alpha$ , is the normalized time between the clock reset and mechanical return, and the speed state,  $\nu$ , is the total mechanical energy at the mechanical return.

We choose to organize the return map,

$$\mathbf{p}_{k+1} = \mathbf{r}(\mathbf{p}_k) = \begin{bmatrix} \nu \Delta H(\mathbf{p}_k) \\ \alpha + \Delta \phi(\mathbf{p}_k) \end{bmatrix}, \quad (\text{C.10})$$

such that its format resembles (2.22). This similarity will help us make ties between this numerical study and the analytic results of Chapter 2. Note that the energy map,  $\Delta H(\mathbf{p}_k)$ , is analogous to the forcing function,  $\rho(\alpha)$ , in (2.26), and the phase map,  $\Delta \phi(\mathbf{p}_k)$ , corresponds to the collective role of the remaining time function,  $\varphi_e(\alpha)$ , in (2.29) and the forcing function,  $\rho(\alpha)$ .

## C.2.4 Local Stability Analysis

The fixed points of the generic return map,  $\mathbf{r}(\mathbf{p})$ , in (C.10) are given by

$$\begin{aligned} \Delta H(\mathbf{p}^*) &= 1, \\ \Delta \phi(\mathbf{p}^*) &= 0. \end{aligned} \quad (\text{C.11})$$

To assess the local stability properties of a fixed point,  $\mathbf{p}^*$ , we will consider the Jacobian of the generic return map,  $\mathbf{r}(\mathbf{p})$ , evaluated at this fixed point,

$$\mathbf{J}(\mathbf{p}^*) = \begin{bmatrix} 1 + \nu^* \cdot \frac{\partial \Delta H(\mathbf{p}^*)}{\partial \nu} \\ \nu^* \cdot \frac{\partial \Delta H(\mathbf{p}^*)}{\partial \alpha} \end{bmatrix} \frac{\partial \Delta \phi(\mathbf{p}^*)}{\partial \nu} \mathbf{1} + \frac{\partial \Delta \phi(\mathbf{p}^*)}{\partial \alpha}. \quad (\text{C.12})$$

whose trace is given by

$$\text{tr} = 2 + \nu^* \frac{\partial \Delta H(\mathbf{p}^*)}{\partial \nu} + \frac{\partial \Delta \phi(\mathbf{p}^*)}{\partial \alpha}.$$

**Lemma 18.** *A fixed point,  $\mathbf{p}^*$ , of the return map,  $\mathbf{r}(\mathbf{p})$ , is unstable if the partial derivatives satisfy*

$$\frac{\partial \Delta H(\mathbf{p}^*)}{\partial \nu} > 0 \quad \frac{\partial \Delta \phi(\mathbf{p}^*)}{\partial \alpha} > 0 \quad (\text{C.13})$$

PROOF. Note that if the trace is greater than two,  $\text{tr} > 2$ , — a sufficient instability condition — then at least one of the eigenvalues of the Jacobian in (C.12) is located outside the unit circle. In particular, if both partial derivatives (C.12) are positive definite, then the trace,  $\text{tr}$ , is guaranteed to be larger than two and the fixed point,  $\mathbf{p}^*$ , is unstable.  $\square$

The linear spring is a degenerate special case. Lemma 19 establishes that clock driven linear spring cannot have no stable behavior.

**Lemma 19.** *The clock driven 1-DOF linear spring is unstable*

PROOF. Recall that the linear spring oscillation period,  $T_m$ , is independent of the total mechanical energy,  $h$ . Hence, actions of the clock controller, which modulates the total mechanical energy,  $h$ , do not affect the mechanical oscillation frequency. Hence, the phases of the clock controller and the mechanical system do not synchronize, and therefore, the coupled system is unstable.  $\square$

Our numerical studies suggest that in the absence of mechanical dissipation clock driven springs, except for the linear spring, are neutrally stable with quasi-periodic limit behavior. In Figure C.4 typical orbits of the Poincaré states for clock driven lossless softening and hardening springs are presented. Recall that in the LPH study we numerically demonstrated that the coupled system is also neutrally stable in the

Figure C.4: Typical neutrally stable orbits of clock driven lossless cyclic systems for three spring types are illustrated: softening spring; linear spring; and hardening spring. For each case we present two plots. The top plots are the generalized position plotted against the time. The markers indicate where the clock discontinuously changes the spring potential. In the bottom plots we depict the Poincaré map. Notice that the neutrally stable fixed points of the softening and hardening springs are located at different delay values. The orbit of the linear spring is not a closed curve but the delay state monotonically increases.

absence of mechanical losses. In fact, there is a Hopf bifurcation as the damping coefficient,  $\mu$ , crosses the zero. In the rest of this discussion we will treat the clock driven lossy springs.

**Conjecture 4.** *The sign of the partial derivatives of the energy map,  $\Delta H(\mathbf{p})$ , and the phase map,  $\Delta\phi(\mathbf{p})$ , are related. Table C.1 lists all the combinations that occur.*

Our numerical studies suggest that the signs of the partial derivatives of the energy map,  $\Delta H(\mathbf{p})$ , and the phase map,  $\Delta\phi(\mathbf{p})$ , are related. Conjecture 4 summarizes those combinations that we observe predominantly. In Figure C.5 and Figure C.6 the reader can find typical plots for the linear, softening and hardening spring cases, respectively. In each figure the partition of the Poincaré space,  $\mathcal{P}$ , according to the sign of each partial derivative is illustrated.

	$\mathcal{P}_i$	$\frac{\partial \Delta H(\mathbf{p})}{\partial \nu}$	$\frac{\partial \Delta H(\mathbf{p})}{\partial \alpha}$	$\frac{\partial \Delta \phi(\mathbf{p})}{\partial \nu}$	$\frac{\partial \Delta \phi(\mathbf{p})}{\partial \alpha}$	$\mathbf{p}^* \in \mathcal{P}_i$
Softening	$\mathcal{P}_1^{soft}$	-	+	-	-	Inconclusive
	$\mathcal{P}_2^{soft}$	+	-	-	+	Unstable
Hardening	$\mathcal{P}_1^{hard}$	+	+	+	+	Unstable
	$\mathcal{P}_2^{hard}$	-	-	+	-	Inconclusive

Table C.1: The sign of partial derivatives are related. Table summarizes those combination of signs that occur for softening and hardening springs. The linear spring is not included in this table since it is guaranteed to be unstable. The last column indicates the limit characteristic of a fixed point if it lives in the corresponding partition of the Poincaré space,  $\mathcal{P}$ .

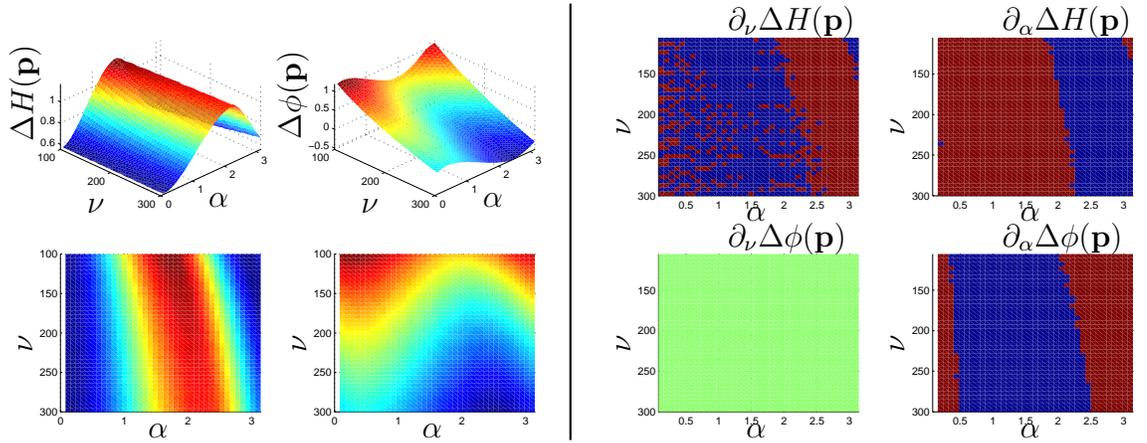


Figure C.5: Numerically computed energy map,  $\Delta H(\mathbf{p})$ , (right) functions for a lossy softening spring,  $U_\xi(x_1) = -\frac{2\xi}{3}\text{sign}(x_1)x_1^{1.5}$ , driven by a 2-cell shape function, (left) and phase map,  $\Delta \phi(\mathbf{p})$ . Top images show the surface plot with perspective where as the bottom ones are color maps of the same functions for easy identification of the gradient.

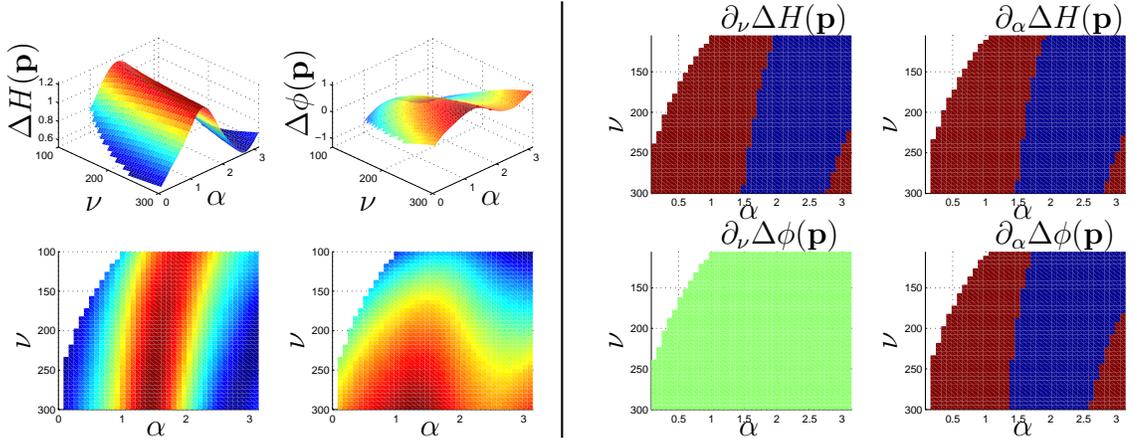


Figure C.6: Numerically computed energy map,  $\Delta H(\mathbf{p})$ , (right) functions for a lossy hardening spring,  $U_\xi(x_1) = -\frac{\xi}{4}x_1^4$ , driven by a 2-cell shape function, (left) and phase map,  $\Delta\phi(\mathbf{p})$ . Top images show the surface plot with perspective where as the bottom ones are color maps of the same functions for easy identification of the gradient.

The observation in Conjecture 4 suggests that the stable fixed points can only occur in certain portions of the Poincaré space,  $\mathcal{P}$ , where one or both of the partial derivatives in (C.13) are negative. The delay states of stable fixed points are restricted into different intervals. Mechanical systems with softening springs can only have stable fixed points where the partial derivative of the energy map with respect to the delay state is positive. Similarly, fixed points of the coupled systems with hardening springs are restricted in the delay interval where the partial derivative of the energy map with respect to the delay is negative.

Our numerical studies suggest that stable limit behavior results for mechanical systems with softening springs when the modulations of the spring parameter occur as the spring transitions from compression to decompression. On the other hand, for hardening springs we observe that the stable limit mechanical behavior corresponds to cases when the spring modulation occurs as the spring transitions from decompression

into compression. Figure C.7 illustrates these two cases.

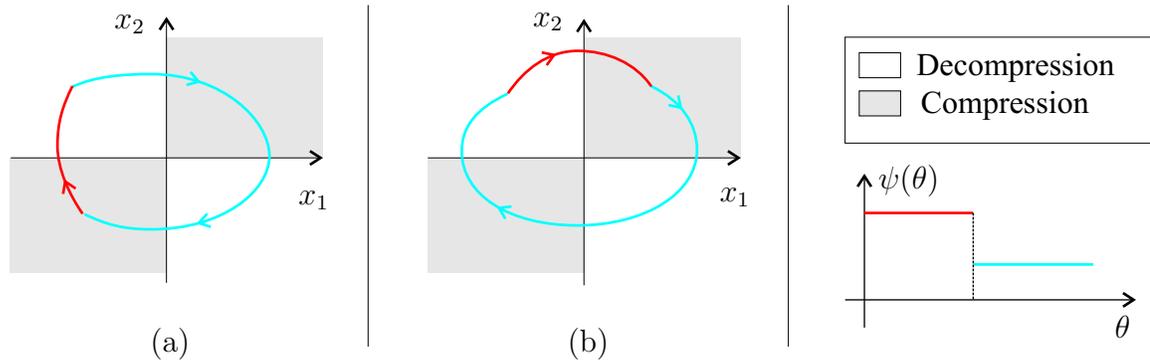


Figure C.7: A conceptual illustration of the two possible flows of the mechanical system at a fixed point of the Poincaré map. In this example we consider a clock controller with a two cell shape function,  $\psi(\theta)$ , shown on the right. Our numerical studies suggest that the softening springs give rise to stable limit behaviors of type (a) and hardening springs result in stable limit behaviors of type (b).

Notice that the LPH in Section 2.1 has a hybrid potential whose overall characteristics fall under softening springs due the contribution of the aerial potential. Our claim in this analysis is confirmed by the results of the LPH analysis in Section 2.4. Recall that the energy map,  $\Delta H(\mathbf{p})$ , is analogous to the forcing function,  $\rho(\alpha)$ . Our results in Section 2.4.4 concluded that the stable fixed points,  $\mathbf{p}^*$ , can only occur where the forcing function,  $\rho(\alpha)$ , has a positive derivative which agrees with our generic claim.

## C.2.5 Hybrid Potentials: Hopper

In the study of hoppers we come across a particular subset of the family of 1-DOF mechanical systems defined in Section C.2.1 where the spring potential is a combination of two potentials, what we will refer as a “hybrid potential.” The linearization

study in Section C.2.4 speculates about the softness of mechanical potential,  $s$ , and the corresponding stability properties of the clock controlled setup. One should note that in the case of hybrid potentials the softness measure,  $s$ , is a function of all the softness measures of the set of potential that make it up.

In the case of clock controlled hopper there are two types of potential laws acting on the system. In the aerial mode,  $\mathcal{A}$ , the system moves under the influence of the gravitational potential,  $U_\xi(x_1) = x_1$ , while in the stance mode  $\mathcal{G}$ , the behavior is taken over by the leg spring potential,  $U_\xi(x_1)$ . The gravitational potential is soft by definition. However, the leg potential can be virtually anywhere along the softness axis. If the leg spring is softening or Hooke's law then obviously the overall potential is softening. On the other hand, if the leg potential is hardening then the overall potential can be softening or hardening depending on the energy level of the mechanical system.

Figure C.8 illustrates the effects of leg spring type on the overall performance of the clock controlled prismatic hopper where we employed (a) softening; (b) Hooke's law; and (c) hardening leg springs. The numerical results suggest that the stability performance of the coupled system gets better as the leg spring becomes more soft.

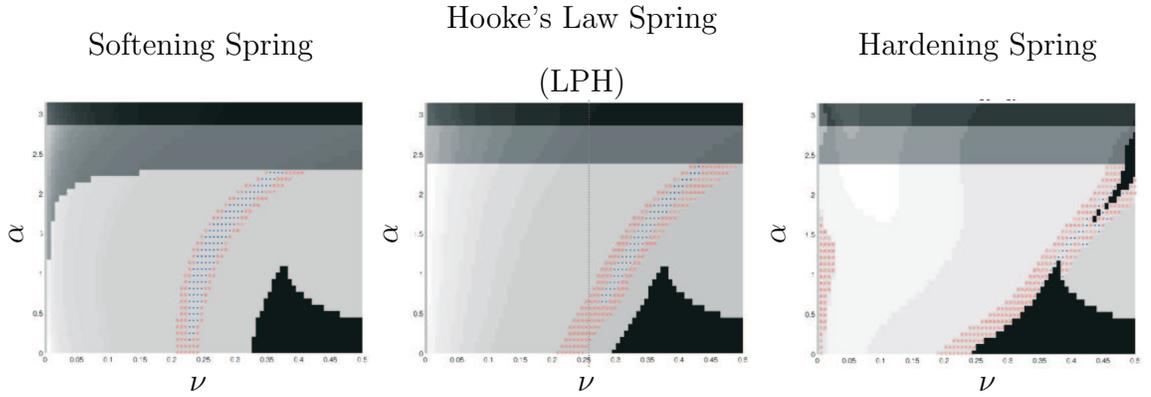


Figure C.8: Mode sequence partitions numerically computed over a 80x40 grid on the Poincaré space,  $\mathcal{P}$ , for prismatic hoppers with three types of springs: (a) a softening spring,  $U_\xi(x_1) = -\frac{2\xi}{3}\text{sign}(x_1)x_1^{1.5}$ ; (b) Hooke's law spring,  $U_\xi(x_1) = -\frac{\xi}{2}x_1^2$ ; and (c) a hardening spring,  $U_\xi(x_1) = -\frac{\xi}{4}x_1^4$ . The grid elements that are in the  $\mathcal{GAG}(1,2)$  mode sequence partition are marked by (blue) dots to indicate convergence to the fixed point,  $\mathbf{p}$ , and by (red) crosses to indicate unstable operation. Hence, region of blue dots approximate the domain of attraction of the fix point in  $\mathcal{GAG}(1,2)$  whose percentage to the entire  $\mathcal{GAG}(1,2)$  partition is (a) %51.31579 for softening spring; (b) %27.01754 for Hooke's law spring; and (c) %9.65517 for hardening spring

# BIBLIOGRAPHY

## BIBLIOGRAPHY

- [1] H. Komsuoglu and D. E. Koditschek. Preliminary analysis of a biologically inspired 1-dof 'clock' stabilized hopper. In *World Multiconference on Systemics, Cybernetics and Informatics (SCI2000)*, volume IX, pages 670–675, Orlando, FL, 2000.
- [2] D. E. Koditschek. Computational neuromechanics: Programming work in biological and mechanical systems. <http://www.eecs.umich.edu/CNM>, 1998.
- [3] M. Cutkosky. Sprawl robots. <http://www-cdr.stanford.edu/biomimetics>, 2000.
- [4] U. Saranli, M. Buehler, and D. E. Koditschek. Design, modeling and preliminary control of a compliant hexapod robot. In *Proceedings of International Conference on Robotics and Automation*, volume 3, pages 2589–2596, San Fransisco, CA, April 2000.
- [5] J. G. Cham, S. A. Bailey, J. E. Clark, R. J. Full, and M. R. Cutkosky. Fast and robust: hexapedal robots via shape deposition manufacturing. *International Journal of Robotics Research*, 21(10):869–883, 2002.
- [6] M. Buehler, U. Saranli, and D. E. Koditschek. Single actuator per leg robotic hexapod, 2002. US Patent 6,481,513 B2.
- [7] J. D. Weingarten, R. E. Groff, and D. E. Koditschek. A framework for the coordination of legged robot gaits. In *IEEE Robotics, Automation and Mechatronics*, Singapore, December 2004. submitted.
- [8] J. D. Weingarten, G. A. D. Lopes, M. Buehler, Groff R. E., and D. E. Koditschek. Automated gait adaptation for legged robots. In *Int. Conf. Robotics and Automation*, New Orleans, USA, 2004. IEEE.
- [9] Saranli U., M. Buehler, and D. E. Koditschek. Rhex - a simple and highly mobile hexapod robot. *International Journal of Robotics Research*, 20(7):616–631, 2001.
- [10] E.Z Moore. Leg design and stair climbing control for the rhex robotic hexapod. Master's thesis, McGill University, November 2001.

- [11] D. Campbell and M. Buehler. Preliminary bounding experiments in a dynamic hexapod. In B. Siciliano and P. Dario, editors, *Experimental Robotics*, chapter VIII, pages 612–621. Springer-Verlag, 2003.
- [12] D. McMordie and M. Buehler. Towards pronking with a hexapod robot. In *Proceedings of 4th International Conference on Climbing and Walking Robots*, Germany, September 2001.
- [13] U. Saranli and D. E. Koditschek. Back flips with a hexapodal robot. In *IEEE International Conference on Robotics and Automation*, 2002.
- [14] H. Komsuoglu. Leaping behavior for a hexapod robot. Technical report, University of Michigan, 2003. In preperation.
- [15] N. Neville and M. Buehler. Towards bipedal running of a six legged robot. In *12th Yale Workshop on Adaptive and Learning Systems*, May 2003.
- [16] K.S. Espenschied, R. D. Quinn, H. J. Chiel, and Beer R. D. Biologically-inspired hexapod robot control. In *Proceedings of the 5th International Symposium on Robotics and Manufacturing: Research, Education and Applications (ISRAM' 94)*, Maui, Hawaii, August 1994.
- [17] R. A. Brooks. A robot that walks; emergent behaviors from a carefully evaolved network. Technical report, MIT AI Lab, February 1989. Technical Report Memo 1091.
- [18] I. E. Sutherland and M. K. Ullner. Footprints in the asphalt. *International Journal of Robotics Research*, 3(2):??, 1984.
- [19] S. Hirose. A study of design and control of a quadruped walking machine. *International Journal of Robotics Research*, 3(2):113–133, 1984.
- [20] D. R. Pugh, E. A. Ribble, V. J. Vohnout, T. E. Bihari, T. M. Walliser, M. R. Patterson, and K. J. Waldron. Technical description of the adaptive suspension vehicle. *International Journal of Robotics Research*, 9(2):24–42, 1990.
- [21] B. Klaassen, R. Linneman, D. Spenneberg, and F. Kirchner. Biomimetic walking robot scorpion: Control and modeling. In *Proceedings of the 9th International Symposium on Intelligent Robotic Systems*, pages 101–108, 2001.
- [22] R. Altendorfer, U. Saranli, H. Komsuoglu, D. E. Koditschek, Jr. H. B. Brown, M. Buehler, N. Moore, D. McMordie, and R. Full. Evidence for spring loaded inverted pendulum running in a hexapod robot. In *International Seminar on Experimental Robotics*, 2000.
- [23] D. E. Koditschek, 2004. Personal Communication.

- [24] Wikipedia. The wheel. <http://en.wikipedia.org/wiki/Wheel>, 2004. An online encyclopedia.
- [25] J. Y. Wong. *Theory of Ground Vehicles*. Wiley and Sons, Inc., 2nd edition edition, 1993.
- [26] R. Dawkins. So why don't animals have wheels? in Sunday Times, November 24 1996.
- [27] H. C. Berg and R. A. Anderson. Bacteria swim by rotating their flagellar filaments. *Nature*, 245:380–382, October 1973.
- [28] R. J. Full, K. Earls, M. Wong, and R. Caldwell. Locomotion like a wheel? *Nature*, 365:495, October 1993.
- [29] R. Dawkins. *Climbing Mount Improbable*. W. W. Northon, New York, 1996. 1st American edition.
- [30] R. E. Ritzmann, R. D. Quinn, J. T. Watson, and S.N. Zill. Insect walking and biorobotics: A relationship with mutual benefits. *BioScience*, 2000.
- [31] K. Hirai, M. Hirose, Y. Haikawa, and T. Takenaka. The development of honda humanoid robot. In *International Conference on Robotics and Automation*, pages 1321–1326, Leuven, Belgium, 1998.
- [32] K. Loffler, M. Gienger, and F. Pfeiffer. Sensor and control design of a dynamically stable biped robot. In *Proceedings of International Conference on Robotics and Automation*, pages 484–490, 2003.
- [33] U. Saranli, W. J. Schwind, and D. E. Koditschek. Toward the control of a multi-jointed, monopod runner. In *International Conference on Robotics and Automation*, volume 3, pages 2676–2682, Leuven, Belgium, 1998.
- [34] J. W. Grizzle, G. Abba, and F. Plestan. Asymptotically stable walking for biped robots: Analysis via systems with impulse effects, February 08 1999.
- [35] J. J. Abbas and R. J. Full. *Neuro-Mechanical Interaction in Cyclic Movements*, volume Biomechanics and Neural Control of Posture and Movement, chapter Section IV: Rhythmic Systems. Springer-Verlag, New York, 2000.
- [36] Aristotle. On the gait of animals. Translated by A. S. L. Farquharson, 350 B.C. [http://classics.mit.edu/Aristotle/gait\\_anim.html](http://classics.mit.edu/Aristotle/gait_anim.html).
- [37] Aristotle. On the motion of animals. Translated by A. S. L. Farquharson. [http://classics.mit.edu/Aristotle/motion\\_animals.html](http://classics.mit.edu/Aristotle/motion_animals.html).
- [38] Galen. De motu musculorum, 150.

- [39] G. Vasari. *Vite de eccellenti Architetti, Pittori, et Scultori Italiani (The Lives of Painters, Sculptors, and Architects)*. 1550.
- [40] A. Doyon. *Jacques Vaucanson, mecanicien de genie*. Presses universitaires de France, 1967.
- [41] G. A. Borelli. *De Motu Animalum (On the Motion of Animals)*. 1679.
- [42] E.-J. Marey. *Animal mechanism: a treatise on terrestrial and aerial locomotion*. D. Appleton and co., 1874.
- [43] E. Muybridge. *Animal Locomotion*. University of Pennsylvania, Philadelphia, 1887.
- [44] E. H. Weber, E. Weber, and E. F. W. Weber. *Die Mechanik Der Menschlichen Gerverzeuge*. 1836.
- [45] R. J. Bachman, G. M. Nelson, W. C. Flannigan, R. D. Quinn, J. T. Watson, A. K. Tryba, and R. E. Ritzmann. Construction of a cockroach-like hexapod robot. In *Eleventh VPI and SU Symposium on Structural Dynamics and Control*, pages 647–654, Blacksburg, VA, May 1997.
- [46] S. Vogel. *Cats' Paws and Catapults: Mechanical Worlds of Nature and People*. Norton, New York, 1998.
- [47] E. Klavins, H. Komsuoglu, R. J. Full, and D. E. Koditschek. *Neurotechnology for Biomimetic Robots*, chapter The Role of Reflexes Versus Central Pattern Generators in Dynamical Legged Locomotion. MIT Press, Boston, 2000.
- [48] M. H. Dickinson, C. T. Farley, R. J. Full, M. A. R. Koehl, R. Kram, and S. Lehman. How animals move: An integrative view. *Science*, 2000.
- [49] A. H. Cohen, S. Rossignol, and S. Grillner. *Neural control of rhythmic movements in vertebrates*. Wilney series in neurobiology. Wiley-Interscience, 1988.
- [50] F. Delcomyn. Neural basis of rhythmic behavior in animals. *Science*, 210:492–498, 1980.
- [51] K. Pearson. The control of walking. *Scientific American*, 235(6):77–86, December 1976.
- [52] R. J. Full. Invertebrate locomotor systems. In W. Dantzler, editor, *The Handbook of Comparative Physiology*, pages 853–930. Oxford University Press, 1997.
- [53] S. Grillner, L. Cangiano, G.-Y. Hu, R. Thompson, R. Hill, and P. Wallen. The intrinsic function of a motor system — from ion channels to networks and behavior. *Brain Research*, 886:224–236, 2000.

- [54] A. H. Cohen. The role of heterarchical control in the evolution of central pattern generators. *Brain Behavior Evolution*, 40:112–124, 1992.
- [55] N. G. Hatsopoulos. Coupling the neural and physical dynamics in rhythmic movements. *Neural Computation*, 8:567–581, 1996.
- [56] K. G. Pearson. Proprioceptive regulation of locomotion. *Current Opinion in Neurobiology*, 1995.
- [57] R. M. Williamson and B. L. Robert. Sensory and motor interactions during movement in the spinal dogfish. *Proceedings of the Royal Society of London Ser. B*, 227:103–119, 1986.
- [58] J. J. Collins and I. N. Stewart. Coupled nonlinear oscillators and the symmetries of animal gaits. *Journal of Nonlinear Science*, 3:349–392, 1993.
- [59] N. Kopell. We got rhythm: Dynamical systems of the nervous system. *American Mathematical Society*, 47(1):6–16, 2000.
- [60] G. S. Medvedev and N. Kopell. Synchronization and transient dynamics in the chains of electrically coupled fitzhugh-nagumo oscillators. *SIAM Journal of Applied Mathematics*, 2001.
- [61] S. Epstein and N. Kopell. A mechanism for resonance tuning in neuromechanical systems. in preparation, 1998.
- [62] A. H. Cohen, G. B. Ermentrout, T. Kiemel, N. Kopell, K. A. Sigvardt, and T. L. Williams. Modelling of intersegmental coordination in the lamprey central pattern generator for locomotion. *Transactions in Neural Science (TINS)*, (11):434–438, 1992.
- [63] N. Bernstein. *The Control and Regulation of Movement*. Pergamon Press Ltd., Oxford, New York, 1967.
- [64] W. J. Schwind and D. E. Koditschek. Approximating the stance map of a 2-dof monopod runner. *Journal of Nonlinear Science*, 10:533–568, 2000.
- [65] C. t. Farley, J. Glasheen, and T. A. McMahon. Running springs: Speed and animal size. *Journal of Experimental Biology*, 185:71–86, 1993.
- [66] R. J. Full and C. T. Farley. *Musculoskeletal Dynamics in Rhythmic Systems - A Comparative Approach to Legged Locomotion*, volume Biomechanics and Neural Control of Posture and Movement, chapter Section IV: Rhythmic Systems. Springer-Verlag, New York, 2000.
- [67] R. Blickhan. The spring-mass model for running and hopping. *Journal of Biomechanics*, 22(11/12):1217–1227, 1989.

- [68] O. Ekeberg. A combined neuronal and mechanical model of fish swimming. *Biological Cybernetics*, 69:363–374, 1993.
- [69] E. Klavins. *Decentralized Phase Regulation of Cyclic Robotic Systems*. Phd, Univerisity of Michigan, 2001.
- [70] R. Rand, A. Cohen, and P. Holmes. Systems of coupled oscillators as models of central pattern generators. In A. H Cohen, S. Rossignol, and S. Grillner, editors, *Neural control of rhythmic movements in vertebrates*.
- [71] R. H. Rand and P. J. Holmes. Bifurcation of periodic motions in two weakly coupled van der pol oscillators. *International Journal of Nonlinear Mechanics*, 15:387–399, 1980.
- [72] D. W. Storti and R. H. Rand. Dynamics of two strongly coupled van der pol oscillators. *International Journal of Nonlinear Mechanics*, 17:143–152, 1982.
- [73] R. Jung, T. Kiemel, and A. Cohen. Dynamical behavior of a neural network model of locomotor control in lamprey. *Journal of Neurophysiology*, 75:1074–1086, 1996.
- [74] R. J. Full, K. Autumn, J. I. Chung, and A. Ahn. Rapid negotiation of rough terrain by the death-head cockroach. *American Zoologist*, 38:81A, 1998.
- [75] H. Cruse. What mechanisms coordinate leg movement in walking anthropods? *Trends in Neuroscience*, 13:15–21, 1990.
- [76] R. McN. Alexander. *Elastic mechanisms in animal movement*. Cambridge University Press, 1988.
- [77] T. A. McMahon. The spring in the human foot. *Nature*, 325, January 8 1987.
- [78] Devin Logan Jindrich. *Stability, maneuverability, and control of rapid hexapodal locomotion*. PhD thesis, University of California, Berkeley, 2001.
- [79] R. McN. Alexander. Three uses for springs in legged locomotion. *The International Journal of Robotics Research*, 9(2):53–61, 1990.
- [80] D. I. Jindrich and R. J. Full. Dynamic stabilization of rapid hexapedal locomotion. *Journal Experimental Biology*, 205:2803–2823, 2002.
- [81] M. H. Raibert and J. K. Hodgins. Legged robots. In R. D. Beer, R. E. Ritzmann, and T. McKenna, editors, *Biological Neural Networks in Invertebrate Neuroethology and Robotics*, pages 319–354. Academic Press, New York, 1993.
- [82] A. G. Fel'dman. Functional tuning of the nervous system during control of movement or maintenance of steady posture. II. controllable parameters of muscle. *Biophysics*, 11:565–578, 1966.

- [83] T. A. McMahon. *Muscles, reflexes and locomotion*. Princeton University Press, 1984.
- [84] C. Farley, R. Blickhan, J Saito, and R. Taylor. Hopping frequency in humans: a test of how springs set stride frequency in bouncing gaits. *Journal of Applied Physiology*, 71(6):2127–2132, 1991.
- [85] D. Ferris, M. Louie, and C. T. Farley. Running in the real world: adjusting leg stiffness for different surfaces. *The Royal Society Proceedings: Biological Sciences*, 265:941–1044, 1998.
- [86] D. P. Ferris, K. Liang, and C. T. Farley. Runners adjust leg stiffness for their first step on a new running surface. *Journal of biomechanics*, 32:787–794, 1999.
- [87] J. J. Craig. *Introduction to robotics: mechanics and control*. Addison-Wesley Publishing Company, 1986.
- [88] M. Raibert. *Legged robots that balance*. MIT Press series in artificial intelligence. MIT Press, Boston, 1986.
- [89] D. E. Koditschek and M. Buehler. Analysis of a simplified hopping robot. Report 8804, Department of Electrical Engineering, Yale University, Princeton, NJ, 1988.
- [90] Z. Li and J. He. An energy perturbation approach to limit cycle analysis in legged locomotion systems. In *Proceedings of 29th IEEE Conference on Decision and Control*, pages 1989–1994, 1990.
- [91] A.F. Vakakis and J.W. Burdick. Chaotic motions in the dynamics of a hopping robot. In *Proceedings of IEEE Conference on Robotics and Automation*, volume 3, pages 1464–1469, May 1990.
- [92] R.T. M’Closkey, J.W. Burdick, and A.F. Vakakis. On the periodic motions of simple hopping robots. In *Proceedings of IEEE Conference on Systems, Man and Cybernetics*, pages 771–777, Nov 1990.
- [93] R. Q Linde. Design, analysis, and control of a low power joint for walking robots, by phasic activation of mckibben muscles. *IEEE Transactions on Robotics and Automation*, 15(4):599–604, 1999.
- [94] G. Zeglin and B. Brown. Control of bow leg hopping robot. 1998.
- [95] M. Ahmadi and M. Buehler. The arl monopod ii running robot: Control and energetics. In *International Conference on Robotics and Automation*, 1999.
- [96] M. Ahmadi and M. Buehler. Preliminary experiments with an actively tuned passive dynamic running robot. In *International Symposium on Experimental Robotics*, 1997.

- [97] W. J. Schwind. *Spring Loaded Inverted Pendulum Running: A Plant Model*. PhD thesis, University of Michigan, 1998.
- [98] W. J. Schwind and D. E. Koditschek. Control of forward velocity for a simplified planar hopping robot. In *International Conference on Robotics and Automation*, pages 691–696, 1995.
- [99] R. J. Full and D. E. Koditschek. Templates and anchors: Neuromechanical hypotheses of legged locomotion. *The Journal of Experimental Biology*, 202(23):3325–3332, 1999.
- [100] M. H. Raibert, M. Chepponis, and B. Brown, Jr. Running on four legs as though they were one. *IEEE Journal of Robotics Automation*, RA-2(2), 1986.
- [101] J. Pratt, P. Dilworth, and G. Pratt. Virtual model control of a bipedal walking robot. In *International Conference on Robotics and Automation*, pages 193–198, Albuquerque, New Mexico, 1997.
- [102] G. M. Nelson and R. D. Quinn. Posture control of a cockroach-like robot. In *International Conference on Robotics and Automation*, pages 157–162, 1998.
- [103] U. Saranli and D. E. Koditschek. Template based control of hexapedal running. In *Proceedings of International Conference on Robotics and Automation*, volume 1, pages 1374–1379, 2003.
- [104] J. Nakanishi, T. fukuda, and D. E. Koditschek. Experimental implementation of a "target dynamics" controller on a two-link brachiating robot. In *International Conference on Robotics and Automation*, pages 787–792, Leuven, Belgium, 1998.
- [105] M. Buehler, D. E. Koditschek, and P. J. Kindlmann. A family of robot control strategies for intermittent dynamical environments. *IEEE Control Systems Magazine*, 272:16–22, February 1990.
- [106] M. Buehler, D. E. Koditschek, and P. J. Kindlmann. Planning and control of robotic juggling and catching tasks. *International Journal of Robotics Research*, 13(2):101–118, April 1994.
- [107] A. Rizzi and D. E. Koditschek. Dynamically dexterous robotics: The two-juggle. In *IEEE International Conference on Robotics and Automation Video Proceedings*, Atlanta, GA, 1993.
- [108] A. A. Rizzi and D. E. Koditschek. Progress in spatial robot juggling. In *International Conference on Robotics and Automation*, 1992.
- [109] R. R. Burridge, A. A. Rizzi, and D. E. Koditschek. Sequential composition of dynamical dexterous robot behaviors. *International Journal of Robotics Research*, 1996.

- [110] H. Komsuoglu, D. McMordie, U. Saranli, N. Moore, M. Buehler, and D. E. Koditschek. Proprioception based behavioral advances in a hexapod robot. In *International Conference on Robotics and Automation*, Seoul, Korea, 2001.
- [111] H. Komsuoglu. Legnet - a distributed wireless network. Technical report, University of Michigan, 2004. In preperation.
- [112] A. Rizzi and D. E. Koditschek. An active visual estimator for dexterous manipulation. *IEEE Transactions in Robotics and Automation*, 12(5):697–713, October 1996.
- [113] G. A. D. Lopes and D. E. Koditschek. Visual registration and navigation using planar features. In *International Conference in Robotics and Automation*, Taipei, Taiwan, 2003. IEEE.
- [114] P-C. Lin, H. Komsuoglu, and D. E. Koditschek. A leg configuration sensory system for dynamical body state estimates in a hexapod robot. In *Proceedings of International Conference on Robotics and Automation*, pages 1391–1396, Taipei, Taiwan, September 2003.
- [115] S. Skaff, G. A. Kantor, D. Maiwand, and A. Rizzi. Inertial navigation and visual line following for a dynamical hexapod robot. In *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, volume 2, pages 1808–1813, October 2003.
- [116] G. A. D. Lopes and D. E. Koditschek. Level sets and stable manifold approximations for perceptually driven nonholonomically constrained navigation. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sendai, Japan, 2004.
- [117] P. R. Ringrose. *Self-Stabilizing Running*. Doctor of philosophy in computer science, Massachusetts Institute of Technology, 1997.
- [118] M. D. Berkemeier and K. V. Desai. A comparison of three approaches for the control of hopping height in legged robots. *International Journal of Robotics Research*, 1998. (submitted).
- [119] P. J. Swanson, R. R. Burridge, and D. E. Koditschek. Global asymptotic stability of a passive juggling strategy: A possible parts feeding method. *Mathematical Problems in Engineering*, 1995.
- [120] J. Guckenheimer and P. Holmes. *Nonlinear Oscillations, Dynamical Systems, and Bifurcations of Vector Fields*, volume 42 of *Applied Mathematical Sciences*. Springer-Verlag, New York, NY, 1983.
- [121] S. Schaal and C. G. Atkeson. Open loop stable control strategies for robot juggling. In *International Conference on Robotics and Automation*, volume 3, pages 913–918, Georgia, Atlanta, 1993.

- [122] A. A. Rizzi. *Dexterous Robot Manipulation*. Phd, Yale University, 1994.
- [123] A. Kuo. The relative roles of feedforward and feedback in the control of rhythmic movements. *Motion Control*, 6:129–145, 2002.
- [124] D. M. Wilson. The origin of the flight-motor command in grasshoppers. In R. F. Reiss, editor, *Neuronal Theory and Modeling*, pages 331–345. Stanford University Press, Stanford, CA, 1964.
- [125] M. Buehler, A. Cocosco, K. Yamazaki, and R. Battaglia. Stable open loop walking in quadruped robots with stick legs. In *International Conference on robotics and Automation*, 1999.
- [126] J. G. Cham, J. K. Karpick, and M. R. Cutkosky. Stride period adaptation for biomimetic running hexapod. *International Journal of Robotics Research*, 23(2), February 2004.
- [127] P. Holmes, 2000. Personal communication.
- [128] P. R. Ringrose. Self-stabilizing running. In *International Conference on Robotics and Automation*, Albuquerque, New Mexico, April 1997.
- [129] M. D. Berkemeier and K. V. Desai. Control of hopping in legged robots using a neural-mechanical approach. In *International Conference on Robotics and Automation*, pages 1895–1701, Detoit, MI, 1999.
- [130] *Calculus: One and Several Variables*. Harcourt Brace Jovanovich Collage Publishers, 1991.