Robot Arm Control Exploiting Natural Dynamics

by

Matthew M. Williamson

M.Eng, University of Oxford (1993)
S.M., Massachusetts Institute of Technology (1995)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

June 1999

Signature of Author . . . .
Department of Electrical Engineering and Computer Science
May 18, 1999

Certified by . . . .
Rodney A. Brooks
Professor of Electrical Engineering and Computer Science, MIT
Thesis Supervisor

Accepted by . . . .
Arthur C. Smith
Chairman, Departmental Committee on Graduate Students

Robot Arm Control Exploiting Natural Dynamics

by
Matthew M. Williamson

Revised version of a thesis submitted to the
Department of Electrical Engineering and Computer Science
on May 18, 1999, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

This thesis presents an approach to robot arm control exploiting natural dynamics. The approach consists of using a compliant arm whose joints are controlled with simple non-linear oscillators. The arm has special actuators which makes it robust to collisions and gives it a smooth compliant motion. The oscillators produce rhythmic commands of the joints of the arm, and feedback of the joint motions is used to modify the oscillator behavior. The oscillators enable the resonant properties of the arm to be exploited to perform a variety of rhythmic and discrete tasks. These tasks include tuning into the resonant frequencies of the arm itself, juggling, turning cranks, playing with a Slinky toy, sawing wood, throwing balls, hammering nails and drumming.

For most of these tasks, the controllers at each joint are completely independent, being coupled by mechanical coupling through the physical arm of the robot. The thesis shows that this mechanical coupling allows the oscillators to automatically adjust their commands to be appropriate for the arm dynamics and the task. This coordination is robust to large changes in the oscillator parameters, and large changes in the dynamic properties of the arm.

As well as providing a wealth of experimental data to support this approach, the thesis also provides a range of analysis tools, both approximate and exact. These can be used to understand and predict the behavior of current implementations, and design new ones. These analysis techniques improve the value of oscillator solutions.

The results in the thesis suggest that the general approach of exploiting natural dynamics is a powerful method for obtaining coordinated dynamic behavior of robot arms.

Thesis Supervisor: Rodney A. Brooks
Title: Professor of Electrical Engineering and Computer Science, MIT

Acknowledgments

I would like to thank my advisor Rodney Brooks for his encouragement and support over these years. He has given me a great deal of intellectual freedom, and his ideas have been a source of inspiration for this work.

I would also like to thank Tomás Lozano-Pérez. A series of discussions with Tomás galvanized me into action and are responsible for this thesis being as comprehensive as is. Thank you.

I would also like to thank Stefan Schaal for originally suggesting that I try using oscillators.

The work in this thesis is from the Cog project, and I would like to thank Brian Scassellati, Cynthia Breazeal and Robert Irie for their support. Matt Marjanović was particularly helpful in the final stages of this thesis with the drumming task, figures, photos and general help. Charlie Kemp has also been kind and supportive. I would also like to thank all the people in the Leg Lab at MIT, particularly Jerry Pratt, Hugh Herr, Dan Paluska, Peter Dilworth, Jianjuen Hu and Gill Pratt. Thank you for your friendship and encouragement.

Annika Pfluger kindly helped with the videos for the defense.

I am also indebted to the officemates that I have had at MIT: Tim Tuttle, Jessica Banks, and Una-May O'Reilly. Una-May proof read many papers and bits of this document, for which I am very grateful.

I would also like to thank Ulf Jonsson, Jorge Gonçalves and Alexandre Megretski for their help with the non-linear analysis section.

On a more personal note, I would like to thank my good friends Joel Minnich, Maura McCarthy and Fiona Percy for their love and support over the years. I would also like to thank Esther Chamberlain for her love, encouragement and patience.

This research was supported in part by the Defense Advanced Research Projects Agency and was monitored by the Office of Naval Research under contract numbers N00014-83-K-0125 and N00014-84-K-0099. The author was supported with a Engineering and Physical Sciences Research Council (EPSRC) scholarship.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Synopsis

This thesis presents an approach to robot arm control exploiting the natural dynamics of the arm and its environment. The idea is to "let the physics do the computation". This results in a system which is versatile, robust, computationally simple and easy to implement.

The approach is motivated by related work in robotics. A number of researchers have presented examples of systems which exploit the natural dynamics of the system to perform tasks in a simple manner. These examples include the passive dynamic walkers of McGeer (1990), the dynamic running machines of Raibert (1986), and the open loop stable juggling machines of Schaal and Atkeson (1993). These approaches either use the dynamics directly, or augment the natural dynamics with simple controllers to provide robust performance.

My approach extends these ideas to a number of rhythmic and discrete tasks using a robot arm. The passive dynamics of the arm are manipulated to align them with the task, so that the natural behavior of the complete system is to perform the task. The motion is achieved using a controller to inject energy into the system.

In particular, my approach consists of using a compliant arm whose joints are controlled using simple non-linear oscillators. The arm has special actuators which make it robust to collisions and give it a smooth compliant motion. The actuators are used to implement low gain position control at each joint. This makes the robot links appear as if they were connected by springs and dampers, giving the whole arm a rich mass-spring behavior. The dynamics can be changed by altering arm posture, stiffness, damping and the manipulated object to match the passive arm dynamics with the task. Non-linear oscillators are used to inject energy and so generate the motion. The oscillators produce rhythmic commands at the arm joints which excite the arm dynamics. The oscillators are adaptive, using feedback from the arm joints to alter the frequency and phase of their outputs. The oscillator behavior is to adjust the commands relative to the arm and task dynamics; this appropriately adds energy to the arm, and produces the required motion.

Using the same compliant arm, oscillators and feedback, a wide variety of tasks have been implemented. These are all dynamic tasks where the spring-like properties of the arm are exploited to produce the motion. The tasks include tuning into the resonant frequencies of the arm itself, juggling, turning cranks in a variety of configurations using both one and two arms, playing with a Slinky toy, sawing wood, throwing balls, hammering nails and even drumming.

For most of these tasks, the oscillators at the joints are completely independent. The oscillators use the mechanical coupling through the arm to coordinate with one another and the task at hand. The variety of coordinated tasks that are possible with such a distributed system highlights the sensitivity of the individual oscillators to the arm dynamics. In fact, the thesis shows that coupling oscillators through the natural dynamics is more robust and more powerful than the more usual method of connecting them into networks.

This precise coordination between the oscillators and the arm comes hand in hand with remarkable robustness. The system is robust in two respects. The same task can be accomplished using a wide variety of oscillator parameters. On the other hand, using a fixed set of oscillator parameters, the task can be accomplished for a wide variety of different system stiffnesses, inertias, and other properties. The robustness is a desirable property for a practical control scheme.

Researchers are sometimes reluctant to use oscillators to control their robots, because they do not have the tools to understand how they work, or how best to set the parameters. The complaint that oscillators of the type used in this thesis are difficult to tune is a common one. To answer these concerns, this thesis develops a variety of oscillator design and analysis techniques. These provide insight into the oscillator behavior, and give clear directions for tuning and design.

The following sections present the approach in more detail, comment on the relationship between this work and more traditional control methods, state the contributions of this thesis, and introduce the following chapters.

1.2 Approach

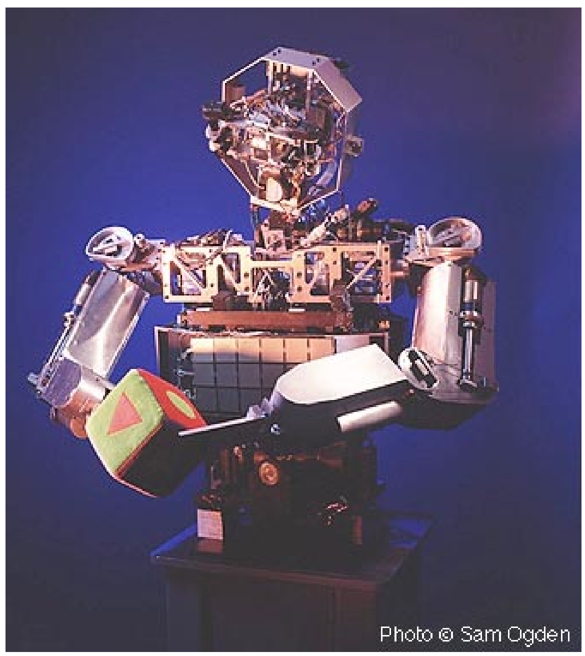

Figure 1-1: The humanoid robot Cog. The two arms used in this thesis are mounted on this robot. Each arm has 6 degrees of freedom arranged in a similar manner to a human arm. The arms are also approximately the same length as a human arm.

This section describes the approach taken in this thesis: the choice and use of a compliant robot arm, and the use of simple nonlinear oscillators to excite and exploit the natural dynamics of the system.

When designing a robot arm, one important consideration is the structural stiffness of the arm. The choice between a stiff robot and a more compliant one rests mainly on the types of tasks envisaged for the robot. For accurate position control and for high bandwidth force control, the robot should be stiff. For tasks which require a robust interaction with an uncertain world, the robot should be designed to be compliant. The dynamical interactions between the links of a compliant arm are more noticeable compared to a stiff arm, which makes the arm's position more difficult to control. For the work in this thesis, a compliant arm was constructed to be used for tasks which required interaction rather than positional accuracy. The extra dynamics are exploited to perform tasks simply.

Two six degree of freedom arms were designed and used for this thesis, both mounted on the humanoid robot Cog (Brooks and Stein, 1994, Brooks et al., 1998), shown in figure 1-1. The arms use series elastic actuators at each joint, an actuator technology which incorporates a physical spring in series with the motor output. (Pratt and Williamson, 1995, Williamson, 1995). By measuring and controlling the deflection of the spring, the force output of the actuator can be controlled. The spring contributes to the smooth force output of the actuator by naturally filtering out the noise and backlash introduced by the gearbox. The spring also makes it easy to ensure that the actuator is passive and therefore stable when interacting with passive environments (Colgate and Hogan, 1989). Another important role for the spring is to absorb shock loads, protecting the motor gearbox from damage. These characteristics greatly contribute to the robustness of the arm design. The design of the arms is described in more detail in Appendix A.

The series elastic actuators provide force control, giving a force output at the joint which is close to the desired force. To control the position of the joint, a low gain proportional-derivative (PD) controller is used, whose output is the desired torque or force at the joints, as shown in figure 1-2. The stiffness and damping at the joints can be varied by changing stiffness K or damping B, and the arm moved around by changing the setpoint θ_v of the controller. The PD control makes the arm behave as if its links are connected by springs and dampers, but because the forces in the joints are accurately controlled by the series elastic actuators, the overall motion is smooth and compliant. The force control bandwidth of the actuators is fairly low, which forces the stiffness of the arm to be low. This gives robustness when interacting with objects.

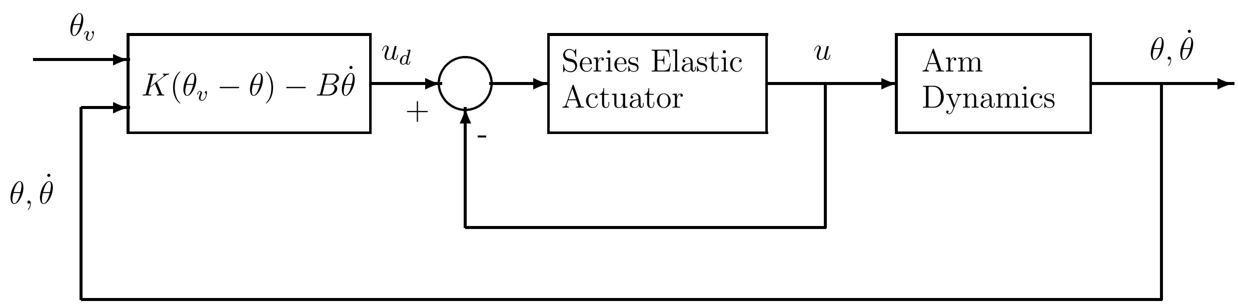

Figure 1-2: The joint level control consists of a position control loop, with stiffness K, damping B, and setpoint θ_v organized around an inner torque control loop provided by the series elastic actuators. The inner loop controls the force output of the joint u to accurately track the desired force u_d. Organizing the control in this fashion results in smooth, compliant motion of the arm, where the overall stiffness and damping can be altered by changing the values of K and B.

To control the arms a set of simple non-linear oscillators are used to alter the setpoints for each joint. The oscillators are models of two biological neurons in mutual inhibition which form a resonant circuit (Matsuoka, 1985). The oscillators respond to the dynamical state of the arm using feedback of either the joint angle or the joint torque. The system is thus tightly coupled, as shown in figure 1-3, because the output from each oscillator drives the joint, and the feedback adjusts the oscillator output. In most of the work for this thesis, there are no connections between the oscillators. Instead the oscillators rely on mechanical coupling between the joints given by the natural dynamics to coordinate with one another.

The oscillators form a dynamical system which, without input, produces a rhythmic output. If a rhythmic input is applied to an oscillator, the oscillator will entrain with that input, producing an output signal of the same frequency. This occurs over a reasonably large range of frequencies and input amplitudes. The entrained behavior is complex, but can be roughly described as “resonant”

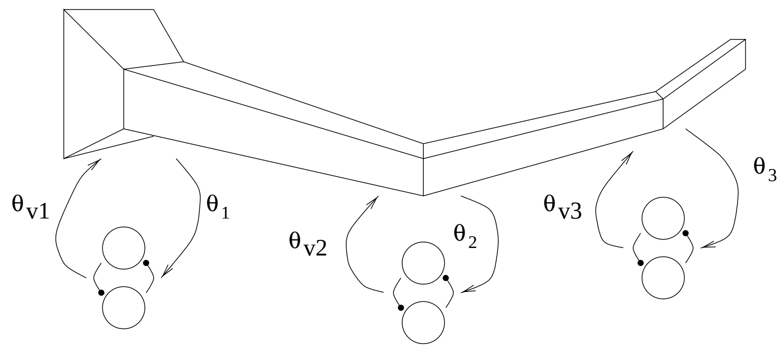

Figure 1-3: The oscillators are connected to each joint of the arm, and are tightly coupled so that the oscillator output θ_{vi} drives the joint, and the oscillator input is either the joint angle θ_i or joint torque u_i. There is generally no explicit connection between the oscillators. The oscillators rely on mechanical coupling through the physical structure of the robot, sensed using feedback to coordinate with one another.

where the oscillator drives the system at a frequency close to resonance, or in a mode similar to the resonant mode of the complete system.

The oscillator behavior makes them suitable as adaptive controllers for the arm. Once the arm has been set up so that the natural dynamics of the coupled arm-task system is appropriate for the task, the oscillator behavior is to respond to the system dynamics and inject energy to produce the required motion. The work in this thesis concentrates on demonstrating, explaining and understanding the behavior of the oscillators in conjunction with the arm natural dynamics. This includes design methods for the oscillator parameters, but does not address how to choose and design the arm parameters: arm posture, stiffness etc.. In most cases the initial arm configuration was chosen by hand, with automatic tuning used to modify that initial configuration. This is described in appendix B.

1.3 Comparisons

The main difference between traditional robot control and the approach taken in this thesis is the role of the robot dynamics. In traditional control, the robot is viewed as a general purpose manipulator which performs tasks independent of the robot configuration. The task is specified in terms of the desired motion (force, position, compliance) of the robot, and the robot control enforces that command. The robot dynamics are generally ignored or canceled, and certainly do not play a part in how the task is planned. The approach taken in this thesis is the opposite: the robot dynamics are crucial for the performance of the task as they determine the range of possible tasks, and also how the tasks are accomplished. The robot dynamics are specified so that the task motion is a passive behavior of the system, and the oscillators are used to inject energy into the arm and so create the motion.

This difference is illustrated in figure 1-4. The task illustrated is that of moving a mass backwards and forwards. In traditional control, the dynamics of the robot are removed, so the equivalent connection between the desired position of the mass x_d, and its actual position x is stiff. The x_d trajectory is required to move the mass backwards and forwards, and the controller needs to overcome the inertial and frictional forces on the mass. If the dynamics of the arm are exploited, represented here by a spring, the situation is somewhat simpler. The natural behavior of the mass is to vibrate on the spring and so move backwards and forwards. The role of the trajectory x_d is now to inject and remove energy to sustain the motion, not create the whole motion.

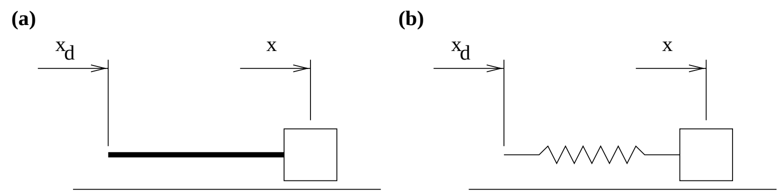

Figure 1-4: Comparison between traditional control **(a)**, and exploiting natural dynamics **(b)** for the task of moving a mass backwards and forwards. Under traditional control the dynamics of the robot are not used. The controller enforces a stiff structure which makes the actual mass position x track closely the desired position x_d. If the robot is made to be compliant, then its dynamics can be exploited to perform the same task in a different manner. The mass will naturally vibrate on the spring of the robot dynamics, and the role of the desired trajectory is to sustain the motion, rather than create the whole motion.

The traditional approach is more general, since the mass can be moved in any arbitrary trajectory x_d. However, for rhythmic tasks, the alternative has some advantages. One consequence of exploiting the dynamics is that the arm needs to be compliant. This has the benefit of giving robust interaction with objects and unexpected collisions. The traditional controllers need to be stiff to reduce tracking errors. This stiffness causes problems in practice: unexpected collisions are not dealt with robustly by high gain position controlled systems, and high gain force control is known to be difficult because of stability issues.

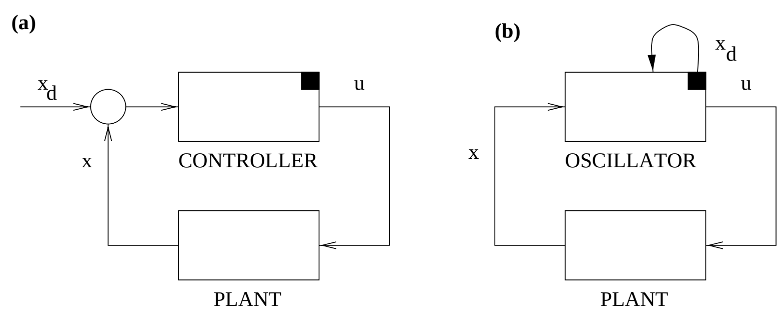

Figure 1-5: Comparison between traditional control **(a)**, and oscillator control **(b)**. In traditional control the plant's state x is controlled to follow the desired state x_d using a controller. This controller could have internal state as indicated by the $\blacksquare$, say from an integral term. The oscillator control has no explicit desired trajectory, this being generated internally by the oscillator dynamics. The oscillator modifies its control action u dependent on the system state x. This has the advantage that a low gain system can be used where the complexity of planning x_d is carried out by the oscillator dynamics rather than by some other system.

The second difference is in terms of the design of the controller, and is illustrated schematically in figure 1-5. While a traditional controller requires a desired trajectory x_d, the oscillator control generates that signal using its internal dynamics. As before, the traditional approach is more general, the oscillator system being restricted to the trajectories that are generated by the oscillator dynamics. Also as before, arranging the control in this way has some practical advantages.

The main advantage of using the oscillator is that the desired trajectory is reactive to the dynamics of the system. Referring to the mass-spring system in figure 1-4, the oscillator can generate a trajectory which complements the motion of the mass, by injecting and removing energy. The x_d generated by the oscillator is reactive since it is calculated within a tight loop, and is synchronized with the system motion since it is generated relative to the state x. These characteristics are achieved without a separate system to calculate x_d, and without the extra sensing, modeling and computation that calculation of x_d would require.

In the case of oscillators controlling multiple joints of the arm, this internal generation of trajectories is even more advantageous. The oscillators are independent, coupled only through the arm dynamics. The trajectories for all the joints are thus generated in a distributed manner with coordination which is correct relative to the motion of the arm. This contrasts with the complexity of the system which would be needed to generate explicit trajectories for all the joints. This difference is accentuated by the versatility of the oscillator system. While calculating x_d for one task is relatively straightforward, repeating this for each joint and each new task would be tedious or require the extra complexity of kinematic modeling and calibration.

A further difference in the robotic case is that the oscillator control system does not deteriorate as the speed of the task increases and the dynamics of the arm become significant. Both position and force control for robots degrades at high speeds because of disturbances from the arm dynamics. If the arm dynamics are aligned with the task, and as the speed increases those dynamics remain aligned with the task, then the oscillator system will be robust to the change in speed.

1.4 Contributions

The contributions of this thesis are

- The demonstration of the versatility of exploiting natural dynamics by showing implementations of a wide variety of different tasks on a real robot arm. These tasks include both rhythmic tasks (e.g. juggling, crank turning, sawing) and discrete tasks (e.g. throwing, hitting, hammering).
- The demonstration of how seemingly complex tasks become simple when cast as resonances of the arm and task. They can then be performed both simply and robustly using oscillators.
- The demonstration that exploiting the natural dynamics using non-linear oscillators is robust and easy to implement.
- The demonstration that the principles behind exploiting the natural dynamics can be applied to create motions which are not completely dictated by the natural dynamics.
- The development of approximate analysis techniques for non-linear oscillators tightly coupled to mechanical systems, and to systems of oscillators coupled through mechanical coupling. These techniques predict the final motion, and so ease design and tuning. They also provide a powerful means of evaluating the robustness and applicability of oscillator based solutions.
- The presentation of exact theoretical results and analysis techniques which describe the oscillator capabilities, performance and stability. These results complement the approximate techniques.

1.5 Thesis outline

The thesis proceeds as follows:

Chapter 2 reviews the relevant literature in robotics. It considers other robotic approaches which exploit natural dynamics and other approaches using oscillators. It also provides evidence taken from neuroscience and motor psychophysics that humans also exploit their natural dynamics.

Chapter 3 introduces oscillator solutions for single degree of freedom motions, driving single links of the robot arm. The oscillator system uses feedback to coordinate its command with the arm, responding on a cycle by cycle basis to the arm state. The solutions are also robust in two respects. A single set of oscillator parameters can accomplish a task for a wide range of system properties (stiffness, inertia, damping etc.), and for a fixed system, a wide range of oscillator parameters can be used to accomplish the task. The chapter introduces analysis techniques which can be used to predict the behavior of oscillator controlled systems, applied to different types of oscillator (e.g. Matsuoka, Van der Pol etc.), and applied to quite complex tasks such as juggling. The analysis is also used to explain the origin of the oscillator robustness.

Chapter 4 examines the behavior of oscillators driving the arm using multiple degrees of freedom, exploiting mechanical interactions through the arm itself to couple and coordinate the arm motion. The analysis tools introduced in chapter 3 are extended to model this multiple degree of freedom case. They indicate that the oscillators find the resonant mode of the underlying mechanical system. The oscillators find this mode automatically and can thus perform complex coordinated tasks in a distributed manner. The main example presented is crank turning. The crank attached to the arm creates a resonant mode which is the correct motion. The oscillators adapt to exactly coordinate with one another to turn the crank, independent of the arm configuration. This coordination behavior is also robust. The oscillators maintain coordination with the task in spite of wide variations in parameters or arm properties.

Chapter 5 moves away from a description of the oscillator properties to consider how to design motions using oscillators while maintaining their robustness properties. The chapter shows that connecting oscillators into networks with fixed connections is not as robust as using coupling them through the natural dynamics. This is because the mechanical coupling "frees up" the oscillator to be synchronized with the task. The explicit connections force the oscillator outputs to be synchronized with each other, rather than the task. The chapter shows how using a single oscillator to drive multiple degrees of freedom or, alternatively, manipulating the dynamics of the arm give an ideal compromise: the desired motion is achieved with robustness.

Chapter 6 returns to the question of analysis and design for oscillator driven systems. The analysis methods presented in chapters 3 and 5 are practically useful but only approximate the system behavior. This chapter presents exact results concerning the boundedness of the oscillator system, its behavior when coupled to systems, and the local stability of oscillator driven limit cycles. It also shows an alternative technique for describing the oscillator and driven systems as piecewise-linear systems which sheds light and intuition on the oscillator behavior. These results are important and useful in terms of design, giving tools for determining the oscillator behavior when coupled to a variety of systems.

Chapter 7 brings the thesis back to the practical implementation of tasks using oscillators. It demonstrates the versatility of oscillator solutions to single motion tasks such as throwing, hitting and hammering, as well as tasks which are a combination of rhythmic and discrete motions, such as drumming. The chapter shows that these tasks are easy to implement with oscillators and that the feedback provides useful behavior—powerful hitting and drumming at a steady rhythm. The chapter also shows the versatility of the oscillators in terms of the sensory signals that they can use to coordinate with the world, using both information related to the arm for throwing, and auditory signals for the drumming application.

Chapter 8 concludes the thesis, and provides suggestions for future work. Following that are a number of appendices, including information about the arm design.

1.6 Note on data in thesis

This thesis includes many results, some of which are simulated and some taken from the real robot. To differentiate the source of the data, the figure captions have been formatted differently. Figures with simulated data are marked $\boxed{\text{SIM}}$ in the caption, and figures with data from the real robot are marked $\boxed{\text{REAL}}$.

Chapter 2

Literature review

2.1 Introduction

This chapter reviews the relevant literature in robotics, previous work using oscillators, and evidence that humans exploit natural dynamics to perform tasks.

2.2 Exploiting natural dynamics

There is a considerable range of literature on robotics research which exploits natural dynamics. The relevant literature has been summarized in table 2.1. This review divides the work into four categories: research which exploits the robot kinematics and statics, the dynamics of the robot itself, the static characteristics of the task, and the dynamic characteristics of the task. The literature is further divided into approaches where passive mechanisms are exploited at the design stage, approaches where dynamics are used to simplify control, and, lastly, where both the design and the control are motivated by exploiting the natural dynamics. The papers themselves are discussed in more detail below.

The idea of designing passive mechanisms that achieve tasks with minimal computation has a long history. Some of the first tasks which fit in this domain were the use of compliance for robot force control, (for a review of the early work see Mason (1982)). Adding compliance makes the task of controlling force significantly easier. The Remote Centered Compliance (Drake, 1977, Whitney, 1982) is a mechanism whose static behavior greatly reduced the difficulty of common assembly tasks. The properties of the mechanism allow robust assembly without requiring any extra sensing, control or computation. Ulrich (1990) is a good example of robot design using passive mechanisms to reduce gravity torques and improve actuator performance.

The passive dynamic walking machines of McGeer (1990), with later developments by Adolfsson et al. (1998), Fowble and Kuo (1996) and Garcia et al. (1998) are good examples of design to exploit robot dynamics. These machines are completely passive, yet produce a coordinated walking powered by gravity. Another example is the masticating robot of Takanobu et al. (1995) where nonlinear springs were used to create a realistic biting action. The open loop control of force in the sheep shearing robots of Trevelyan (1992) is also relevant. The open loop behavior of the cutter kept it a fixed distance above the skin of the sheep, so achieving the goal of only cutting the fleece of the sheep.

Robot design where the dynamics of the task are exploited include the work of Shannon, whose passive juggling robot could juggle three balls completely passively (this robot is described in Schaal and Atkeson (1993)), the gymnastic robots of Playter and Raibert (1994) and Playter (1994), which exploited the springy dynamics of the arms of a doll to maintain stability while performing flips, and the drumming robot described by Hajian et al. (1997). The drumming robot used a low impedance drumstick holder and exploited the bounce of the stick on the drum to produce high frequency drum rolls.

	Design	**Control**	**Both**
Robot Statics	Compliance: Mason (1982) Remote Centered Compliance: Drake (1977), Whitney (1982) Mechanical design: Ulrich (1990)		
Robot Dynamics	Passive Dynamic Walkers: McGeer (1990), Adolfsson et al. (1998), Garcia et al. (1998), Fowble and Kuo (1996) Mechanical design: Ulrich (1990) Chewing robot: Takanobu et al. (1995) Sheep shearing: Trevelyan (1992)	Walking robot: Pratt and Pratt (1998) Humanoid: Kuniyoshi and Nagakubo (1997) Control: Slotine (1988)	Running robots: Raibert (1986), Playter and Raibert (1992) Internal inertia: Brown (1994) Almost passive walking: der Linde (1998) Arm tasks: THIS THESIS
Task Statics		Force/position control: Raibert and Craig (1981) Impedance control: Hogan (1985a) Constrained tasks: Niemeyer and Slotine (1997)	
Task Dynamics	Shannon juggler: Schaal and Atkeson (1993) Robot gymnastics: Playter and Raibert (1994), Playter (1994) Drumming: Hajian et al. (1997)	Juggling: Rizzi and Koditschek (1994), Schaal and Atkeson (1993), Mason and Lynch (1993)	Juggling: Lynch et al. (1998) Arm tasks: THIS THESIS

Table 2.1: Table of literature which exploits natural dynamics

The second column in table 2.1 describes work which exploits natural dynamics in the control systems for robots. The work in this area which corresponds to exploiting the robot dynamics includes the walking robot of Pratt and Pratt (1998) which exploits passive leg dynamics to enable the use of a simple controller for walking, and the work of Kuniyoshi and Nagakubo (1997) which exploits the dynamic behavior of a simulated humanoid robot in the task of standing up from a sitting position. Slotine (1988) argued for the use of physical concepts such as energy in designing control systems

Work in this area which exploits the static aspects of tasks include hybrid force/position control Raibert and Craig (1981), where position and force constraints were used to modify the robot motion in orthogonal directions. Impedance control, Hogan (1985a), generalizes this approach by manipulating the impedance of the robot endpoint. A more recent example of exploiting constraints is the work of Niemeyer and Slotine (1997) where the constraint is used to determine the motion, so that the robot follows the direction of least resistance.

Task dynamics have been exploited for control by a number of authors, a common application being juggling a ball on a paddle. Rizzi and Koditschek (1994) used controllers that moved the robot paddle in a scaled mirror image of the ball motion. Schaal and Atkeson (1993) presented a variety of schemes which exploited the dynamics of the juggling task itself to achieve juggling with very simple controllers. For example, juggling of one ball could be achieved using a paddle driven with a constant frequency sine wave. The ball natural behavior is to entrain with the paddle motion, giving a stable juggling action with no explicit control. This contrasts with the control scheme of Rizzi and Koditschek (1994), where the paddle position was carefully controlled dependent on the ball state. Mason and Lynch (1993) described another juggling application, using a hand-coded controller to control the throw of a juggling club.

The final column of table 2.1 concerns approaches where the natural dynamics are exploited both in terms of design and control. A good example of this approach in terms of robot design are the running robots of Raibert (1986). Raibert used hydraulic actuators which gave his running robots a springy bouncing behavior, and then used three simple controllers to modify the speed, height and pitch stability of the bouncing motion. This approach yielded robots which could run fast as well as do gymnastic maneuvers (Playter and Raibert, 1992). Another example is the work of Brown (1994), who designed a robot to exploits its internal inertia for tasks such as tugging on a stuck door and weight lifting. The walking robot of der Linde (1998) augments a passive dynamic walker with small amounts of control to achieve an energy efficient walking robot.

An example of exploiting task dynamics is the robotic juggling robot presented by Lynch et al. (1998). This robot used a carefully designed hand and controller to manipulate the ball.

The work presented in this thesis fits into the final column of table 2.1. The arm used is specifically designed to be compliant and have spring-like dynamics. These dynamics are then exploited by the oscillator control. The work fits into two rows of the table: the robot dynamics are exploited to create efficient movement, and the task dynamics are exploited to coordinate the various joints of the arm with the task itself. The work extends previous research both in terms of the complexity and variety of applications demonstrated.

2.3 Behavior-based robotics

The work in this thesis is motivated by research on behavior-based robotics (Brooks, 1986). This has enjoyed considerable success in the mobile robotics community for tasks such as obstacle avoidance and navigation. The ideas have not been well accepted in the manipulation community and there are just a few examples of the ideas being applied to robot manipulation. For example, Smithers and Malcolm (1988) used reactive controllers with a traditional planner for assembly tasks.

The main reason for the lack of transfer is the complex kinematics of robot arms and tasks as compared to mobile robots. There is not the same obvious mapping from sensors to actions (e.g. if the arm hits something it is not obvious how to move away from the obstacle). This has lead to solutions where the motion of the joints was directly mapped to the task (e.g. Asteroth et al. (1992)), very simple arms are used (Connell, 1988), or simple combinations of joints are used (Williamson, 1996, Marjanović et al., 1996). Connell (1994) decomposed the problem into a number of sensor-driven primitives which he combined to pick objects from cluttered work areas. Gershon (1990) described the decomposition of a sewing task into independent modules which were combined to robustly respond to the unpredictable behavior of the fabric. Combining behaviors is also more complicated for arms than for mobile robots. Beccari and Stramigioli (1998) suggested using impedance control to combine behaviors for assembly tasks.

Most behavior-based approaches build successively more complex behavior by combining simple, reactive controllers. The work in this thesis shows how a range of behaviors can be obtained with a single reactive controller. However, some characteristics of the oscillator control are analogous to behavior-based methods. The oscillator is tightly coupled to the environment, and relies on interaction with the environment to create complex behavior. The system also produces complex behavior without requiring explicit models of the controller or the environment.

2.4 Previous work using oscillators

A table summarizing the previous work using oscillators is included in table 2.2. The table compares the literature on a number of axes: whether the papers describe learning, the type of oscillator used, the task described, whether networks of oscillators are used, whether feedback is used, and whether the paper describes a simulated or real implementation. The papers are roughly ordered and grouped in terms of task complexity.

The first group of papers describe learning algorithms to set the connection strengths between groups of oscillators. Doya and Yoshizawa (1989) describes back-propagation for recurrent neural networks, Ermentrout and Kopell (1994) give a method to tune connection strengths in models of biologically plausible neurons, and Nishii (1998) provides methods for tuning connection strengths for oscillators described by phase dynamics. The task in all these papers was to produce a desired frequency and phase output from an oscillatory network. The work in this thesis for the most part exploits the natural dynamics to coordinate multiple oscillators and, so, does not use learning of connection strengths.

The next group of papers exploit the dynamics of an oscillator network to generate gait patterns similar to those observed in quadrapedal locomotion. The idea behind using an oscillator network for this is to produce a variety of different gaits under the control of one parameter. The network provides the coordination of the limbs, and the overall characteristics are controlled by other parameters. An ideal implementation would respond to an increase in frequency by changing from walking to trotting and then to a galloping rhythm. Kimura et al. (1993) presented an oscillator model for insect walking gaits, and Pribe et al. (1997) presented a different oscillator model (Elias-Grossberg) which also had gait changing properties. Collins and Richmond (1994) argued that the details of the network connectivity, rather than the detailed dynamics of the individual oscillators are important to produce patterns and transitions. Zielińska (1996) used a Van der Pol oscillator to generate the rhythm for a two joint leg, and Cohen et al. (1982) showed how coupling between simple oscillators could be used to generate an undulatory locomotion pattern. None of these papers take the dynamics of the task into account, which differs from the work in this thesis. The thesis work requires the natural dynamics in order to coordinate multiple degrees of freedom, and uses that dynamics to remove the need for explicit networks of oscillators.

The next two papers considered address the oscillator property of resonance tuning which is automatically driving a mechanical system at its resonant frequency. An oscillator system with this property was presented by Hatsopoulos et al. (1992), who subsequently used this task to argue for the use of oscillators coupled to systems (Hatsopoulos, 1996). This thesis further demonstrates the

Paper	**Learning**	**Type of oscillator**	**Task**	**Connections**	**Feedback**	**Real**
Doya and Yoshizawa (1989)	√	S	Matching phases	√		
Ermentrout and Kopell (1994)	√	C	Matching phases	√		
Nishii (1998)	√	S	Matching phases	√		
Kimura et al. (1993)		C	Gait pattern	√	√	
Pribe et al. (1997)		C	Gait pattern	√		
Collins and Richmond (1994)		VDP, C	Gait pattern	√		
Zielińska (1996)		VDP	Gait pattern	√	√	
Cohen et al. (1982)		S	Undulatory pattern	√	√	
Hatsopoulos et al. (1992)		C	Resonance tuning	Single DOF	√	
Hatsopoulos (1996)		VDP	Resonance tuning	Single DOF	√	
Doya and Yoshizawa (1992)	√	C	Simple locomotion	√	√	
Nishii (1995)	√	S	1 DOF hopping	Single DOF	√	
Wadden and Ekeberg (1998)		C	Single leg	√	√	
Taga et al. (1991)		M	Bipedal walking	√	√	
Taga (1995a,b)		M	Bipedal walking	√	√	
Miyakoshi et al. (1998)		M	3D Bipedal walking	√	√	
Kimura et al. (1998)		M	Quadrapedal walking	√	√	√
Ekeberg (1993)		C	Undulatory locomotion	√	√	
Ijspeert et al. (1998)	GA	C	Undulatory locomotion	√	√	
Hollerbach (1981)		S	Handwriting			
Miyakoshi et al. (1994)		M	Juggling	Single DOF	√	
Schaal and Sternad (1998)		M	Arm control	√	√	
THIS THESIS		M	Arm control		√	√

Table 2.2: Table of literature of previous work using oscillators. The type of oscillator is indicated with S, meaning simple phase dynamics, VDP meaning Van der Pol oscillator, M meaning Matsuoka oscillator, and C meaning a different oscillator with complex dynamics.

power of coupling between oscillators and mechanical systems and extends their application to a broad range of tasks.

A large proportion of the oscillator literature is devoted to locomotion partly because it is an obvious rhythmic task and partly because of the biological origins of central pattern generators for locomotion (see section 2.6). Nearly all the work in this field is simulated, with rare examples of implementations on real robots. The next group of papers give a number of applications in locomotion.

There is a considerable range of applications, perhaps the most simple being the rolling example given by Doya and Yoshizawa (1992). An oscillator was tuned to move a mass on a disk, so causing it to roll.[1] Single degree of freedom control of a hopping robot was described by Nishii (1995), and the control of a single leg was described by Wadden and Ekeberg (1998). A task with more reasonable complexity was addressed by Taga et al. (1991). Taga presented a bipedal robot which walked using a network of neural oscillators to control the joints of the robot. The oscillators used feedback from the biped joint angles to entrain the oscillators with the motion. The biped was robust to perturbations and could walk up inclines due to this coupled oscillator-system dynamics. Later papers have added impedance control and finite state machines to control the walking (Taga, 1995a,b), and the latest papers describe obstacle avoidance using the dynamics of the oscillator system (Taga, 1998), and an application to 3D walking (Miyakoshi et al., 1998). These papers are responsible for the popularity of the Matsuoka oscillator used throughout this thesis and have inspired at least one implementation on a real robot (Kimura et al., 1998).

Undulatory locomotion is another application domain, inspired by work in the neuroscience and mathematical community on the lamprey (e.g. Cohen et al. (1982)). This primitive fish appears to use a set of coupled oscillators to produce its undulatory motion. Ekeberg (1993) described a computer model of lamprey swimming which included turning and noted the use of feedback to the oscillators to overcome perturbations. Ijspeert et al. (1998) used genetic algorithms to evolve a similar creature, although he also added legs to produce a creature that could both walk and swim.

As with previous work, the emphasis in these approaches is the connections between oscillators to produce the required coordination patterns for locomotion. Feedback has a minor and poorly understood role making the systems more robust to system perturbations. The work in this thesis has the opposite emphasis. Feedback is taken to be most important, being used to provide coordination and robustness, and connections are shown to be more difficult to tune, and not able to provide motion which adapts to the dynamics of the task at hand.

There are very few implementations of oscillators for arm control. Hollerbach (1981) described using simple oscillators to model and generate handwriting, and Miyakoshi et al. (1994) presented a simulated juggling task. Schaal and Sternad (1998) suggested a general framework of using oscillators to generate rhythmic movements of arms. The work presented in this thesis extends the use of oscillators to considerably more complex tasks than described in the literature. It also differs from most of the literature in that it is implemented on a real robot.

2.5 Oscillator analysis

Given the range of literature on oscillators, it is surprising that there is not more practical analysis of oscillator behavior. The work of Matsuoka (1985, 1987) examined stability of networks of neurons, but did not address the effect of feedback. The effect of inputs to neurons is often described using phase-response curves (e.g. Canavier et al. (1997), Dror et al. (1999)), although these appear to be appropriate for oscillators which are accurate models of biological neurons, producing spikes rather than sinusoidal-like outputs. Other work on neurons is more mathematical (for example Terman

[1] This task was presented as a learning problem although simply connecting a Matsuoka oscillator to the same system resulted in rolling with minimal tuning. The reason for this may be the difference in the dynamic richness of the Matsuoka oscillator and the oscillator used by Doya and Yoshizawa (1992).

et al. (1998) and other work by Kopell) and has the aim of understanding the dynamics of biological neurons. The analysis presented in this thesis provides accurate, simple analysis which is geared towards understanding the behavior of oscillators coupled to systems, as well as providing practical information for tuning purposes.

2.6 Biological central pattern generator evidence

There is considerable evidence for oscillator or central pattern generator control of human and animal limbs. The most striking evidence comes from observing the behavior of the hind legs of spinalized cats, when placed on a treadmill. The action of the treadmill on the legs causes the legs to walk, even though there is no direct spinal control (see Rossignol and Dubuc (1994), Rossignol (1996) for reviews). The legs coordinate with each other, and produce a fairly normal looking stepping pattern. There is also evidence that humans have a similar behavior (see Barbeau and Rossignol (1994) for a review).

Another thread of research is the physiology of invertebrates, where pattern generators are thought to be responsible for locomotion of insects, lampreys etc.. Arshavsky et al. (1991) and Getting (1988) provide reviews of the literature in this field, and Pearson (1993) relates the invertebrate and vertebrate literature.

2.7 Human arm control

There is considerable evidence that humans exploit the dynamics of their bodies in the way that we perform tasks, move, and perceive objects.

For example, we organize our kinematics in many different ways to be appropriate for different tasks. When playing pool, our whole bodies are arranged to put the cue in line of sight of our eyes, and the mechanical control is directed from one isolated joint. When writing or performing a delicate task, we often brace our hands on a hard surface, isolating and removing excess degrees of freedom, and aligning those that are left with the axes of the task. Another example is using our skeleton rather than our muscles to carry loads. Hogan (1985b) described the effect of postural changes and musculature on the mechanical properties of arms.

There is evidence that human exploit the compliant spring-like dynamics of our limbs in a variety of ways. The compliance itself appears to be passive (Mussa-Ivaldi et al., 1985) and thus appropriate for interacting with objects in a stable manner. The spring-like properties of our muscles and tendons are exploited during running (Alexander, 1990). The resonant properties of our limbs are also exploited during ordinary movement, Herr (1993) and Hatsopoulos and Warren (1996) showing that the preferred frequency of swinging arms is the resonant frequency of the limb and its environment, determined by measuring the most "comfortable" swinging frequency when the arm dynamics had been augmented by either extra springs or added weights. In addition Herr (1993) provided experimental data that the comfortable frequency corresponded to a minimum of work. Even babies seem to be able to respond to the resonant properties of the environment, quickly finding the resonant frequency of bouncing supports (Goldfield et al., 1993). Bingham et al. (1989) showed that when throwing objects a sequence of postures is used which exploits the dynamical coupling between the limb segments. He also showed that the mass properties of the object are estimated in order to tune muscle stiffnesses for a powerful throw. The idea that when the arm performs a task it is a "smart" mechanism, adjusting its passive properties for the task has been suggested by Saltzman and Kelso (1987) and Bingham et al. (1991) amongst others.

There is some evidence from development (Thelen et al., 1992) and also from motor learning (Schneider et al., 1989) that the goal of learning is to correct the timing and strength of the muscle forces to complement the natural dynamics. Thus the interaction forces between the limb segments are harnessed to produce the movement, or in the words of Bernstein:

> ...the secret of co-ordination lies not only in not wasting superfluous force in extinguishing reactive phenomena but, on the contrary, in employing the latter in such a way as to employ active muscle forces only in the capacity of complementary force. (Bernstein, 1967, p. 109)

The interplay between explicit control and natural dynamics is a subject of debate. If the natural dynamics were the most important aspect of arm control, one might expect the same task to be carried out differently in different parts of the workspace because of the different mechanisms used. This does not seem to be the case, there being a remarkable similarity between arm motions in different parts of the workspace, and tasks performed at different scales. For example, signatures tend to look the same when written very small, very large or even when written with other body parts such as one's foot. Humans also tend to produce roughly straight movements over the workspace of the arm (Morasso, 1981, Atkeson and Hollerbach, 1985). Although some researchers argue that many of the qualities of arm motion can be explained either by complex muscle models (Gribble et al., 1998), or neural network models (Massone and Myres, 1996), the similarity of movements suggests that a fairly complex mechanism is controlling the arm to compensate for the dynamics of the arm itself.

A complex mechanism is required because the dynamics of human arms are considerable, their stiffness being low and the stiffness decreasing during movement (Zajac, 1989, Bennett et al., 1992). Experiments where the arm dynamics are perturbed (e.g. Gandolfo et al. (1996)) or where the perception of the arm is perturbed (e.g. Wolpert et al. (1995)) both show humans rapidly adjusting their movement towards straight line motion. This further suggests active control of the arm. Whether this control is an explicit controller or whether it is something more aligned with Bernstein ideas is also a subject of debate.

One experiment which illustrates the trade off between control and exploiting dynamics is the study on obstacle avoidance by Sabes and Jordan (1997). Sabes found that the closest distance between the object and the arm correlated well with the arm's inertial properties at that point. The arm was controlled to avoid the obstacle, but the "safe distance" was determined by the arm dynamic properties.

The dynamics of objects also appear to be important in perception. Many perceptual properties can be explained by humans sensing the inertia of objects (Turvey and Carello, 1995). For example, humans are good at estimating the lengths of objects they manipulate, properties that correlate well with the objects inertia. (Pagano and Turvey, 1995) suggested that this mechanism is used to determine limb orientation. Weights were added to the arm which caused subjects to point not with the axis of their arm but with the inertial axis of their arm and the added weights.

2.8 Conclusion

This chapter has situated the work in this thesis in the relevant literature. The approach taken exploits the natural dynamics of the robot and the task, and does so both in terms of the robot design, but also in the choice of control. The work extends previous work on oscillators by addressing a wide range of complex tasks, and by being implemented on a real robot. The emphasis on feedback rather than networks of oscillators also differentiates this work from most oscillator research. The analysis tools presented in this thesis also extend the available analysis techniques.

The chapter has also shown that using oscillators has a firm biological basis, and that exploiting natural dynamics is an approach which humans take when performing particular tasks, and may also be important from general motion.

Chapter 3

Single degree of freedom motions

This chapter introduces the oscillators which are used throughout this thesis. It describes their governing equations, and how they are coupled to the joints of the arm. The chapter then introduces an analysis technique using describing functions which is useful for understanding the non-linear behavior of the oscillators. This analysis technique is accurate, and can be used in a number of ways. The analysis makes clear the effect of parameters on the overall system behavior, which facilitates tuning of parameters. It also shows the inherent robustness of the oscillator properties to parameter and system changes.

The analysis is not restricted to the Matsuoka oscillator and can be equally applied to other oscillator types. The chapter shows its application to the well known Van der Pol oscillator. Neither is the analysis restricted to simple tasks, the chapter describing its application to the design of oscillator driven juggling. This is implemented on the robot.

The chapter concludes by stressing the coordination between the oscillator dynamics and the arm dynamics, and the sensitivity of the oscillators to the exact motion of the arms.

3.1 Introduction

The oscillator used throughout this thesis was originally analyzed by Matsuoka (1985, 1987). It consists of two simulated neurons in mutual inhibition as shown in figure 3-1. The oscillator model approximates the envelope of the firing rate of a real biological neuron with self-inhibition. The model has 4 state variables, governed by the following equations:

$$\begin{aligned}
\tau_1 \dot{x}_1 &= c - x_1 - \beta v_1 - \gamma [x_2]^+ - \Sigma_j h_j [g_j]^+ && (3.1)\\
\tau_2 \dot{v}_1 &= [x_1]^+ - v_1 && (3.2)\\
\tau_1 \dot{x}_2 &= c - x_2 - \beta v_2 - \gamma [x_1]^+ - \Sigma_j h_j [g_j]^- && (3.3)\\
\tau_2 \dot{v}_2 &= [x_2]^+ - v_2 && (3.4)\\
y_i &= [x_i]^+ = \max(x_i, 0) && (3.5)\\
y_{out} &= [x_1]^+ - [x_2]^+ = y_1 - y_2 && (3.6)
\end{aligned}$$

Each simulated neuron is governed by two equations, neuron 1 by (3.1) and (3.2) and neuron 2 by (3.3) and (3.4). The variables x_1, x_2 represent the firing rate of the neurons, and the v_1, v_2 variables are internal states representing the self-inhibition. The outputs of the neurons are taken to be the positive parts of the firing rates, with $y_1 = [x_1]^+$ and $y_2 = [x_2]^+$. The output of the oscillator is taken to be the difference of these signals (3.6). The neurons inhibit one another, with the output of the other neuron appearing in the update for the firing rate, as in the term $-\gamma[x_2]^+$ in (3.1).

The parameters of the oscillator are β, γ which are constant,[1] the constant or tonic parameter c, which determines the amplitude of the oscillator output (defined as A_n), and the two time constants

[1] The values used throughout the thesis were $\beta = 2, \gamma = 2$. Working parameter ranges are described by Matsuoka (1985)

τ_1, τ_2 which determine the frequency and shape of the output. For stable oscillations τ_1/τ_2 should be in the range 0.1–0.5.[2] Keeping the ratio τ_1/τ_2 constant makes the natural frequency of the oscillator ω_n (the frequency of the oscillator without an input) proportional to $1/\tau_1$, as shown in figure 3-2. Figure 3-3 shows a typical output from the oscillator.

Inputs are applied to the oscillator through the variables g_j, weighted by gains h_j. The inputs are arranged to always inhibit the neuron, applying the positive part $[g_j]^+$ to one neuron, and the negative part $[g_j]^- = \max(-g_j, 0)$ to the other. To remove offsets in g_j a high pass filter is used to remove the DC component.

When no input is applied to the oscillator, it oscillates at a natural frequency w_n determined by the time constants τ_1, τ_2, with a fixed amplitude defined by the tonic c, as shown in figure 3-2. If an oscillatory input is applied, the oscillator entrains the input, producing an output at the same frequency as the input. This entrainment behavior is illustrated in figure 3-4 which shows the output of the oscillator as the size of the input signal is increased. The oscillator tends to entrain very quickly, within one cycle in the lower graph in figure 3-4. The oscillator can lock onto input frequencies over a wide range of frequencies and sizes of inputs as illustrated in Figure 3-5. The figure shows the minimum input required to frequency lock the oscillator as a function of frequency. The plot was obtained by varying the input magnitude and comparing the oscillator frequency (taken as the frequency with the maximum magnitude in a Fourier transform of the output), with the input frequency. The entrainment range is large, in this case $w_n = 7$ rad/s, and the range is 1.5 to 35 rad/s.

For most of the work in this thesis, the oscillator is connected to the joints of the robot arm. As described in chapter 1, the position of each joint is controlled using a simple proportional-derivative control law, making the commanded torque at the ith joint

$$u_{di} = k_i(\theta_{vi} - \theta_i) - b_i\dot{\theta}_i \tag{3.7}$$

where k_i is the stiffness of the joint, b_i the damping, θ_i the joint angle and θ_{vi} the equilibrium point. The oscillator is connected to the arm using the oscillator output to control the setpoint for the joint with an output gain h_o and offset θ_{pi}

$$\theta_{vi} = h_o y_{out} + \theta_{pi} \tag{3.8}$$

and connecting either the joint angle θ_i or the joint torque u_i to the input g of the oscillator. The offsets in these signals are removed using a high pass filter. Coupling the oscillator in this way results in a number of useful properties. These are described in the following sections.

3.2 Analysis methodology

The oscillator is a non-linear system, and its behavior when coupled to the mass-spring system of a robot joint is complex. The final motion depends on the interaction between the oscillator dynamics and the system dynamics. This section describes an analysis method that can be used to understand the nature of this interaction, and shed light on the oscillator behavior, stability, tuning and robustness. The analysis is based on expressing the oscillator and the driven system as frequency responses, and determining conditions for oscillation. This technique is known as describing function analysis (Gelb and Vander Velde, 1968, Slotine and Li, 1991, Khalil, 1996).

The coupled system is illustrated in figure 3-6, where the driven system or plant is assumed to be linear with transfer function $G(j\omega)$, where $j = +\sqrt{-1}$ and ω is the frequency. The oscillator is non-linear and so does not have a frequency response. An approximation to the frequency response can be made by sampling over a range of frequencies and input amplitudes. Writing this linearized

[2] The ratio used throughout the thesis was $\tau_1/\tau_2 = 0.5$.

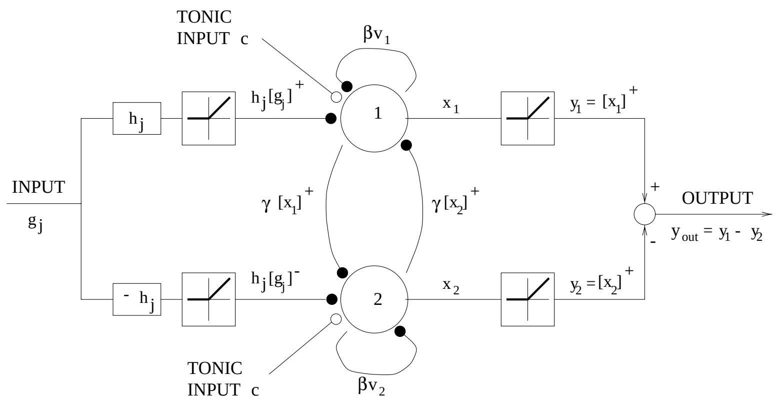

Figure 3-1: Schematic of the oscillator. The oscillator equations simulate two neurons in mutual inhibition as shown here. Black circles correspond to inhibitory connections, open to excitatory. The mutual inhibition is through the $\gamma[x_i]^+$ connections, and the βv_i connections correspond to self-inhibition. The input g_j is weighted by a gain h_j, and then split into positive and negative parts. The positive part inhibits neuron 1, and the negative part neuron 2. The output of each neuron y_i is taken to be the positive part of the firing rate x_i, and the output of the oscillator as a whole is the difference of the two outputs.

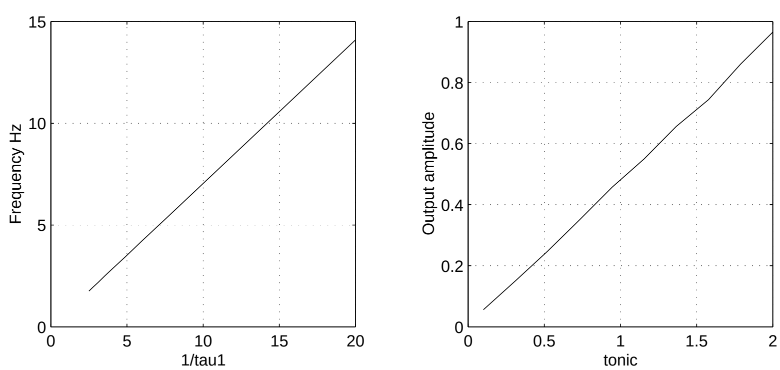

Figure 3-2: $\boxed{\text{SIM}}$ Oscillator behavior under changing tonic c and time constant τ_1. The left hand figure shows the output frequency of the oscillator w_n plotted against $1/\tau_1$, and the right hand graph shows the amplitude of the oscillator output A_n plotted against the tonic excitation c. For this example $\tau_1/\tau_2 = 0.5, \beta = 2.5, \gamma = 2.5$.

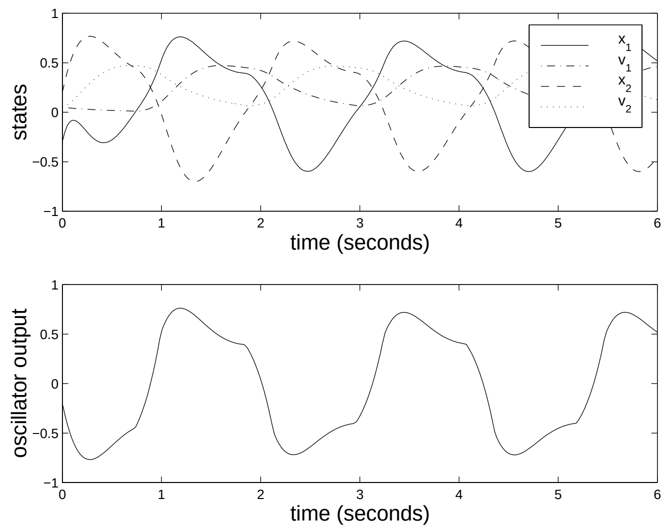

Figure 3-3: $\boxed{\text{SIM}}$ A sample output from the oscillator. The top graph shows the variation with time of the states of the oscillator x_1, x_2, v_1, v_2 during normal operation. The bottom graph shows the output of the oscillator $y_{out} = [x_1]^+ - [x_2]^+$. For this example $\tau_1 = 0.25, \tau_2 = 0.5, c = 1.5, \beta = 2.5, \gamma = 2.5$.

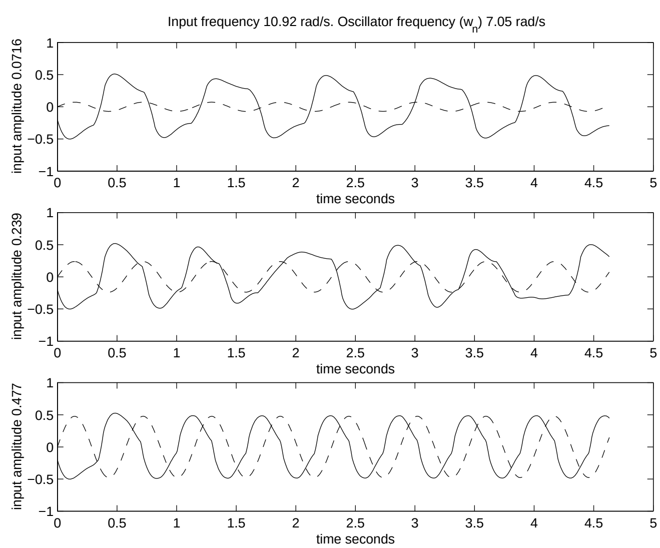

Figure 3-4: $\boxed{\text{SIM}}$ Figure showing effect of increasing input signal. For small input (top graph), the oscillator is not entrained, and oscillates at its endogenous frequency w_n. In the middle graph, the input is larger, and the oscillator is almost entrained, but slips every couple of cycles. The lower graph shows the oscillator locked onto the frequency of the input. For this example $c = 1.0, \tau_1 = 0.1, \tau_2 = 0.2$.

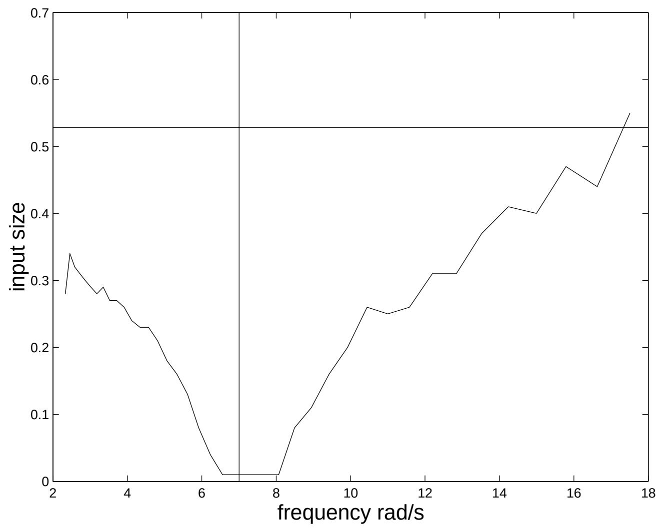

Figure 3-5: $\boxed{\text{SIM}}$ Minimum input for entrainment. The figure shows the minimum input signal required for entrainment of the oscillator. The oscillator natural amplitude and frequency are given by the horizontal and vertical lines respectively. For this example $c = 1.0, \tau_1 = 0.1, \tau_2 = 0.2$.

response at frequency ω and input amplitude A as $N(j\omega, A)$, then the condition for steady state oscillation is that the loop gain is unity:

$$N(j\omega, A)G(j\omega) = 1 \tag{3.9}$$

Calculating N and solving this equation for the frequency and amplitude of the limit cycle solution (ω_f, A_f) provides much insight into the capabilities and tuning of oscillator driven systems. The following section describes how to calculate N for a particular choice of oscillator, and how to solve this equation to calculate the final solution.

3.3 Oscillator describing function

The oscillator describing function $N(j\omega, A)$ expresses the gain and phase difference between input and output of the oscillator as a function of input amplitude and frequency. This can be calculated by applying an input $g_j = A\sin(\omega t)$, and measuring the output y_{out}. If the oscillator is entrained, the frequencies of input and output are the same and a Fourier transform can be used to calculate an approximation to the gain and phase of the oscillator. By applying inputs over a range of frequencies and amplitudes and removing points where the oscillator is not entrained, N can be calculated.[3]

Figure 3-7 shows N plotted as a Bode plot. The plots show the variation of gain and phase with frequency. The multiple lines correspond to different values of A, the arrow referring to the direction of increasing A. Changes in amplitude affect the gain (gain decreasing with increasing A), but not the phase, which remains roughly independent of A. For this oscillator, the gain is actually proportional to $1/A$, as the output amplitude is approximately constant. This is because the input is arranged to always inhibit the neurons. If the input is applied without the max operator, the output

[3] It is also possible to calculate this analytically, using some advanced techniques in describing function analysis, taken from Gelb and Vander Velde (1968).

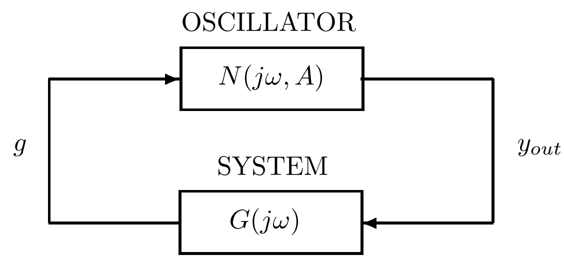

Figure 3-6: System schematic. The figure shows an oscillator tightly coupled to a system. The oscillator output y_{out} drives the input of the system, whose output g is the oscillator input. The analysis of the system proceeds by looking at the frequency responses of each system and calculating the condition for oscillation. At a frequency ω, and input amplitude A, the linearized frequency response of the oscillator can be written as $N(j\omega, A)$, where $j = +\sqrt{-1}$. The frequency response of the system is $G(j\omega)$. There will only be stable oscillations when the loop gain is unity, or $N(j\omega, A)G(j\omega) = 1$.

amplitude becomes a function of A, and the range of possible gains is reduced (see Appendix C for more details).

A common situation is the oscillator driving a single link of the arm. In this case the plant or driven system can be modeled as a mass m actuated through a spring of stiffness k, and a damper b (where k and b are the stiffness and damping terms in the joint level control, equation (3.7)). Ignoring the actuator dynamics, the dynamics of the link are given by the equation

$$m\ddot{\theta} + b\dot{\theta} + k\theta = k\theta_v \tag{3.10}$$

where θ is the position of the link, and θ_v is the desired position. Since the link is connected in the usual way to the oscillator, with $y_{out} = \theta_v$, and $g_j = \theta$, the transfer function for the plant is between θ_v and θ:

$$G(j\omega) = k/(k - m\omega^2 + j\omega b) \tag{3.11}$$

The condition for limit cycles (equation (3.9)) can now be solved graphically by plotting $G(j\omega)$ and $1/N(j\omega, A)$ on the complex plane and finding points where the graphs intersect at the same frequency. This plot is shown in figure 3-8. There are many intersection points, but there is only one where the frequencies of both transfer functions are the same. Since the exact solution is unlikely to be at the points sampled when calculating N and G, a simple interpolation routine is used to find the limit cycle solution. The intersection point (marked with a $\bullet$), determines both the frequency and amplitude of the limit cycle solution. In this case $\omega_f = 8.9$ and $A_f = 0.52$.

3.4 Predictive accuracy

Describing function analysis is an approximate method, and relies on the assumption that the system is well described by the lowest frequency harmonic (Slotine and Li, 1991, Khalil, 1996). This requires firstly that the output of the oscillator not contain many higher harmonics, and secondly that the gain of the linear system be much lower at these harmonics, i.e. that the linear system be a low pass filter. The Fourier transform of the oscillator output is shown in figure 3-9. Most of the energy is located around the natural frequency of the oscillator. Since the mass-spring system has a low pass characteristic, it is appropriate to use the describing function analysis in this case.

The accuracy of the prediction can be found using simulation. Figure 3-10 shows the predicted and measured (from a simulation of the system) frequency and amplitude as the stiffness of the spring

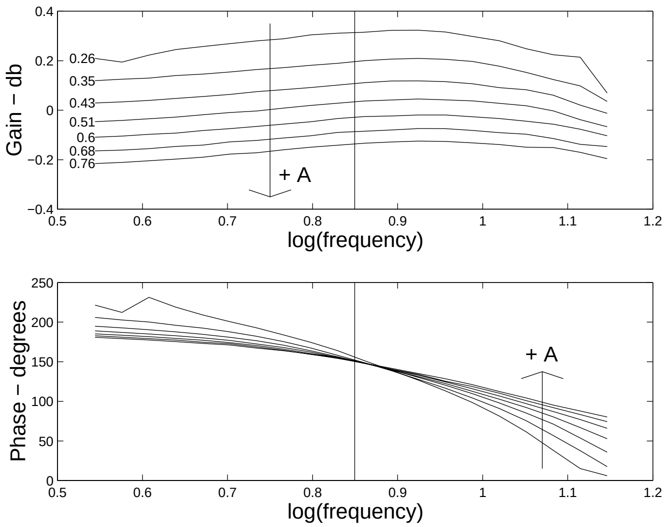

Figure 3-7: $\boxed{\text{SIM}}$ Oscillator Bode plot. The top graph shows the gain $|N(j\omega, A)|$ plotted against frequency, and the lower plot the phase. The multiple lines correspond to different values of input amplitude A as indicated by the numbers and arrows on the plot. The gain is inversely related to A and is roughly constant with frequency. The phase is less dependent on A, and reduces from around 180° to 60° as the frequency is increased. The natural frequency of the oscillator is indicated by the vertical line.

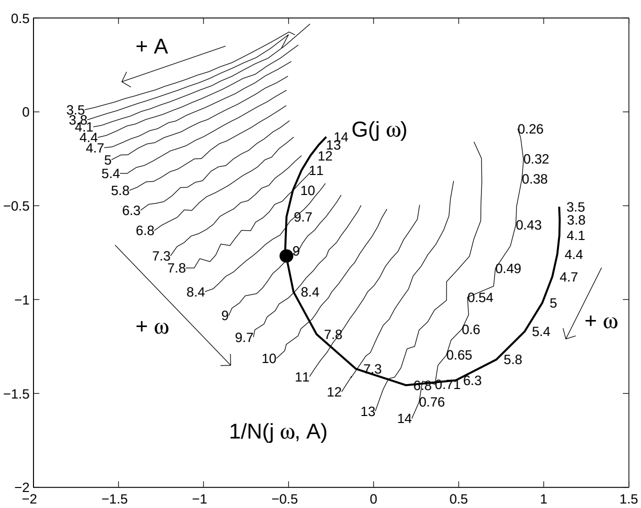

Figure 3-8: $\boxed{\text{SIM}}$ Plot of $G(j\omega)$ and $1/N(j\omega, A)$ in the complex plane. $G(j\omega)$ is the thick line, with numbers indicating frequency. The lines for $1/N(j\omega, A)$ correspond to constant frequency, with amplitude A increasing along each line as indicated by the numbers and arrow. There is a limit cycle solution at the point where the two graphs intersect at the same frequency. The intersection point is indicated by a filled circle, at $\omega_f = 8.9$ and $A_f = 0.52$. This plot was produced for $\tau_1 = 0.1, \tau_2 = 0.2, c = 1, h_j = 1, k = 20, m = 0.4, b = 2$.

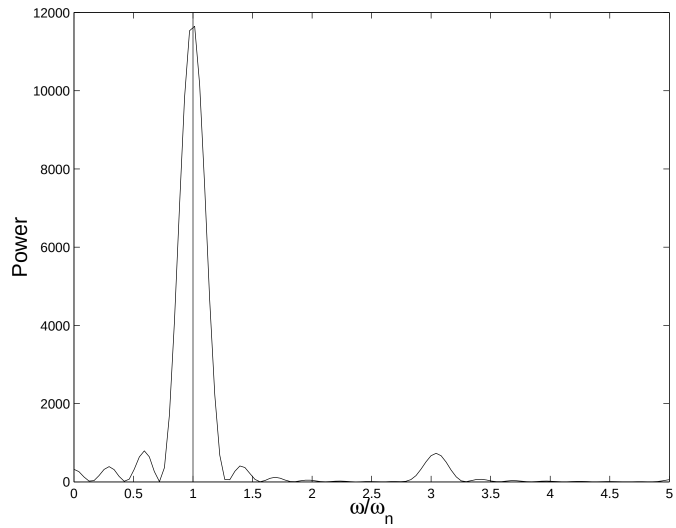

Figure 3-9: $\boxed{\text{SIM}}$ Power density spectrum for the Matsuoka oscillator. The plot shows power plotted against frequency, expressed as a function of the oscillator natural frequency ω_n. The plot shows a clear peak at the natural frequency, and a much smaller one at $\omega = 3\omega_n$. The sinusoidal nature of the oscillator output confirms the accuracy of the describing function approximation.

is altered. Changing stiffness alters the shape of the $G(j\omega)$ plot, and so alters the final solution found by the oscillator. The figure shows that the prediction is accurate, with possible errors coming from the calculation of $N(jw, A)$, as well as the estimation of the frequency and amplitude of the mass motion.

3.5 Local stability

The plot of $G(j\omega)$ and $1/N(j\omega, A)$ can also be used to determine the stability of the final motion, using an extended version of the Nyquist Criterion. The criterion is taken from (Slotine and Li, 1991, p. 186):

Limit Cycle Criterion: *Each intersection of the curve $G(j\omega)$ and the curve $1/N(A)$ corresponds to a limit cycle. If points near the intersection and along the increasing–A side of the curve $1/N(A)$ are not encircled by the curve $G(j\omega)$, then the corresponding limit cycle is stable. Otherwise the limit cycle is unstable.*

In the case of the Matsuoka oscillator driving a mass, examining figure 3-8 shows that for low A, the points are encircled by $G(j\omega)$, but, as A increases, the points move into an un-encircled region. This corresponds to locally stable behavior. Since the analysis only considers the steady state solutions and not the transient behavior, it cannot predict global stability. Experimentally, the cycles are also observed to be stable. Figure 3-11 shows a phase plot for the motion of a robot arm link, for a number of different initial conditions. The system converges to the limit cycle very rapidly, within one cycle in this case.

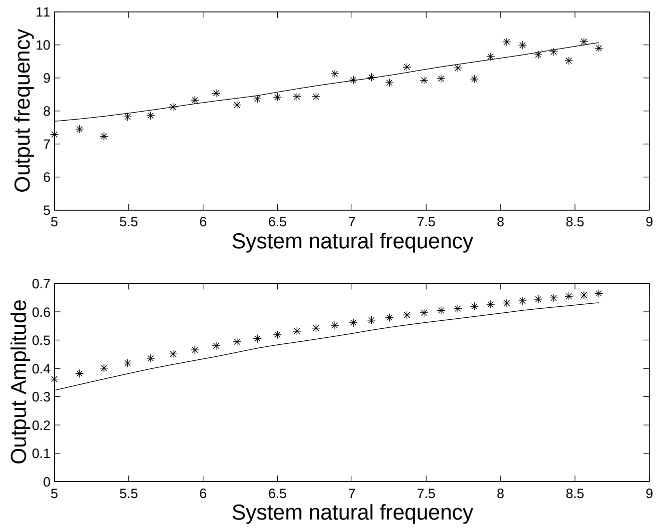

Figure 3-10: $\boxed{\text{SIM}}$ Predictive accuracy. The plot shows the prediction (line) and measured (*) frequency (top plot) and amplitude (lower plot) for an oscillator driving a mass, as the stiffness of the spring was varied, so varying the natural frequency of the system. The accuracy is good, with a constant offset error for amplitude. The error in frequency is partly due to noise in the frequency measurement of the simulated mass motion.

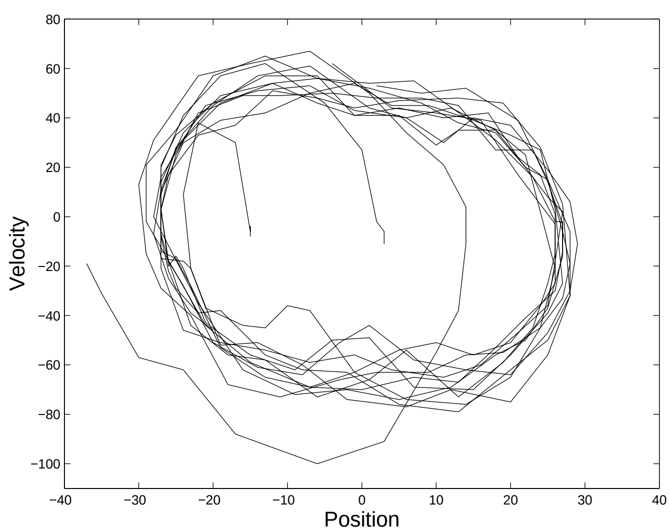

Figure 3-11: $\boxed{\text{REAL}}$ Phase plot of robot motion. The plot shows velocity v. position for an oscillator driving a mass, recorded from one link of a robot arm. Shown are three traces overlaid starting from three different initial conditions. The three traces all converge to the limit cycle, indicative of the anecdotal observation that all initial conditions converge to the limit cycle.

3.6 Effect of parameter changes (a): Design

The oscillator has a number of parameters, time constants, input gains, tonic excitations etc. which affect the final motion of the coupled system. The describing function analysis sheds light on the effect of changes in those parameters, by looking at the effect on the plots $G(j\omega)$ and $N(j\omega, A)$. The effect of most parameters is to scale or shift N while preserving its shape. This means that design can be accomplished without requiring the laborious recalculation of N.

Frequency and amplitude parameters. Altering the natural frequency of the oscillator does not alter the shape of N, but simply scales the frequency of the plot. The shape of the plot is the same if the frequencies are measured with respect to the oscillator natural frequency ω_n. Since for the Matsuoka oscillator $\omega_n \propto 1/\tau_1$, changes in time constants can be easily accounted for.

Similarly, the amplitude of the oscillator is proportional to the parameter c. Changing c increases the amplitude of the oscillator, but does not not change N, except that larger inputs are needed to entrain the system. The shape of the plot remains the same when the input is scaled by the oscillator natural amplitude A_n.

Input and output gains. In general, the oscillator input is multiplied by a gain h_j, and the output also has an output gain h_o, as shown in figure 3-12. The effect of these gains is to change $N(j\omega, A)$ to $h_j h_o N(j\omega, h_j g_j)$, which alters the magnitude but not the phase. Changing the gains thus changes the overall size of N, and can be used to move the plots over one another, or to change the limit cycle solution. Rather than recalculating N, the effect of the gains can be lumped with G, i.e. $G'(j\omega) = h_j h_o G(j\omega)$, which is easily recalculated.

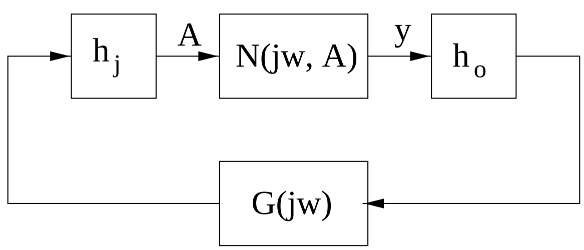

Figure 3-12: If N has an input gain h_j, and an output gain h_o, the overall gain of the oscillator becomes $h_j h_o N(j\omega, A)$. Since N depends on A, i.e. amplitude after the gain h_j, it makes sense to combine the gains with the plant transfer function G, i.e. $G'(j\omega) = h_j h_o G(j\omega)$.

Input and output variables. Choosing a particular oscillator type fixes $N(j\omega, A)$, while choosing outputs and inputs fixes $G(j\omega)$. As shown above, changing oscillator parameters can scale the size of N, but not alter its phase. Thus G needs to be chosen so that the two plots can intersect (there are phases that match). It is then relatively easy to find gains to place the plots over one another. This aspect of tuning is also discussed in appendix B.

Taking for example the mass-spring system of equation (3.10). If instead of position θ, the velocity of the mass $\dot{\theta}$ is used as input, the frequency response becomes

$$G'(j\omega) = j\omega G(j\omega) = j\omega k/(k - m\omega^2 + j\omega b) \tag{3.12}$$

which is the same as before, only rotated 90° counterclockwise, and scaled by ω.