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

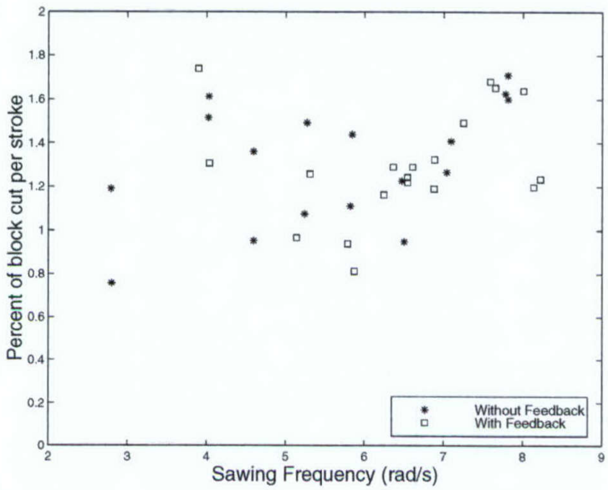

Figure 5-12: REAL Percentage of wood cut per stroke plotted against average sawing frequency. The time and number of strokes to cut a 0.5"by 1.5" block of wood was measured, and the amount of wood cut per stroke calculated as 1/number-of-strokes. The plot shows results using feedback ($*$), and without feedback ($\Box$). There is very little difference between the two methods, although the data is extremely noisy.

5.4 Case (c): Modifying the natural dynamics

Chapter 4 showed that the periodic solutions found by sets of uncoupled oscillators corresponded to the natural modes of the underlying dynamics. This section considers the design approach of artificially altering the natural dynamics of the arm to manipulate the position of the resonant mode. This method exploits the mode-finding properties of the oscillators, and so has greater robustness to parameter changes than the previous two connection methods. On the other hand, it requires the design and implementation of the extra forces.

The dynamics of the arm can be easily altered by applying extra forces to the joints of the arm, perhaps computed from a defined potential field. If a potential field is defined as a function of joint angles, e.g., $V(\Theta)$, the extra torques at the joint can be calculated from the relation:

$$\tau_{potential} = -\frac{\partial V}{\partial \theta_i} \tag{5.5}$$

This can be easily added to the usual joint torque:

$$\tau_i = k_i(\theta_{vi} - \theta_i) - b_i\dot{\theta}_i + \tau_{potential} \tag{5.6}$$

The oscillator can then be used to drive the system, and the final motion is expected to be a resonant mode of the natural dynamics of the system plus the extra potential field. Unlike cases (a) and (b), this method exploits the oscillator properties to find the coordination between the joints and so should have the same kind of robustness and self-organizing properties as in the tasks described in chapter 4.

This method offers the promise of creating phase differences between the joints which are dependent on the task, rather than being restricted to $\pm\pi$. It also creates the possibility of producing motions which are more complicated than that possible using connections. This is because the

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

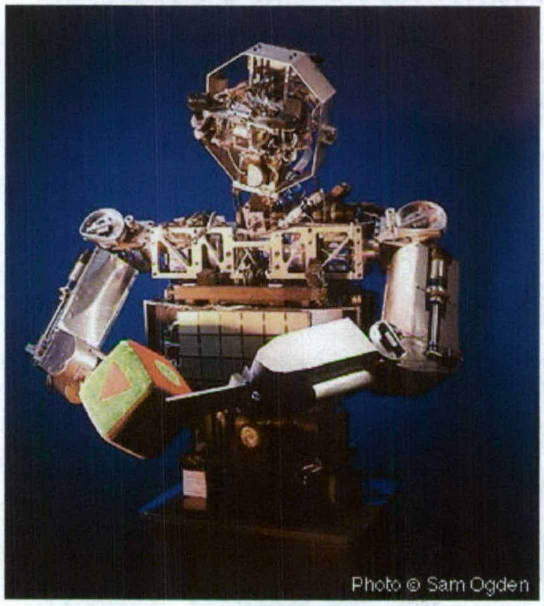

Figure 1-1: The humanoid robot Cog. The two arms used in this thesis are mounted on this robot. Each arm has 6 degrees of freedom arranged in a similar manner to a human arm. The arms are also approximately the same length as a human arm.

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.

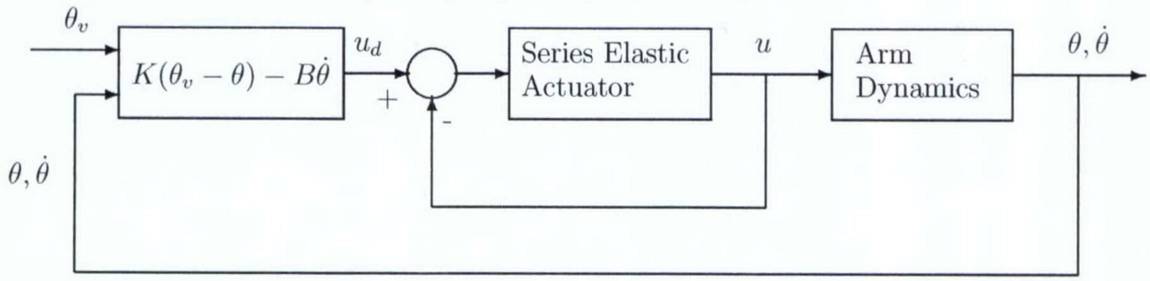

Figure 1-2: The joint level control consists of a position control loop, with stiffness K, damping B, and setpoint θ_v organized around an inner torque control loop provided by the series elastic actuators. The inner loop controls the force output of the joint u to accurately track the desired force 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"

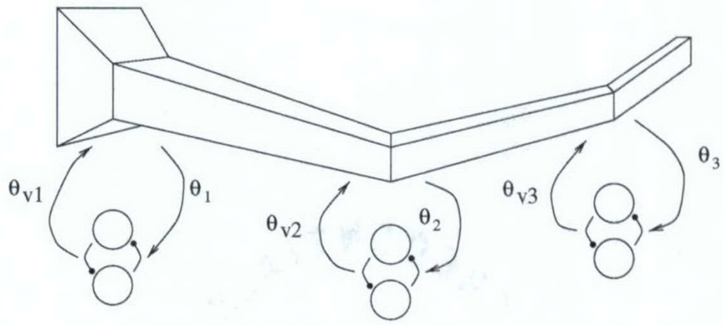

Figure 1-3: The oscillators are connected to each joint of the arm, and are tightly coupled so that the oscillator output θ_{vi} drives the joint, and the oscillator input is either the joint angle θ_i or joint torque u_i. There is generally no explicit connection between the oscillators. The oscillators 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.

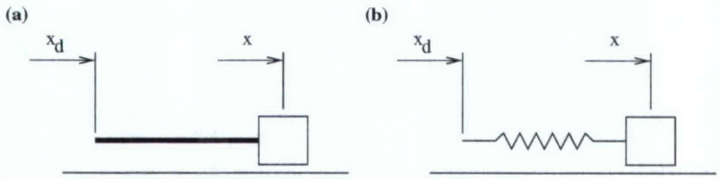

Figure 1-4: Comparison between traditional control **(a)**, and exploiting natural dynamics **(b)** for the task of moving a mass backwards and forwards. Under traditional control the dynamics of the robot are not used. The controller enforces a stiff structure which makes the actual mass position x 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.

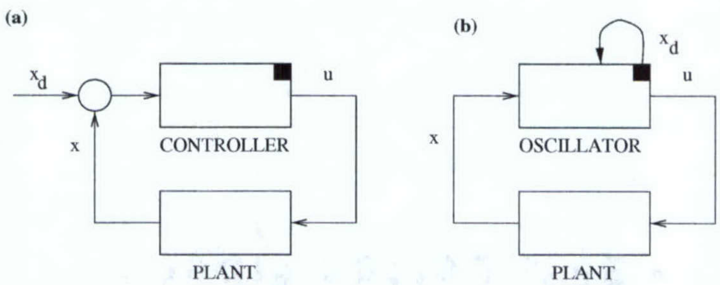

Figure 1-5: Comparison between traditional control **(a)**, and oscillator control **(b)**. In traditional control the plant's state x is controlled to follow the desired state x_d using a controller. This controller could have internal state as indicated by the ■, say from an integral term. The oscillator 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 [SIM] in the caption, and figures with data from the real robot are marked [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:

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

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 $\tau_1/\tau_2 = 0.5$.

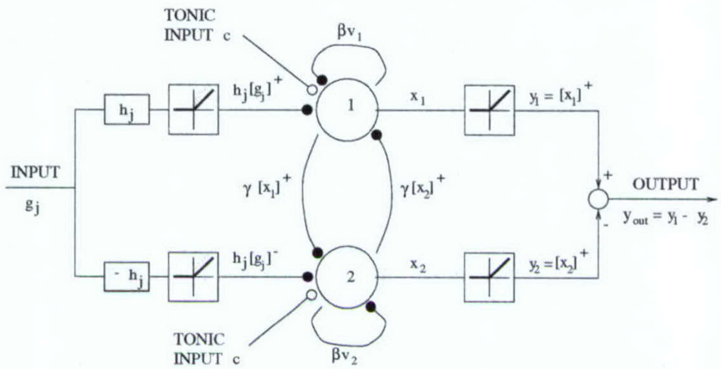

Figure 3-1: Schematic of the oscillator. The oscillator equations simulate two neurons in mutual inhibition as shown here. Black circles correspond to inhibitory connections, open to excitatory. The mutual inhibition is through the $\gamma[x_i]^+$ connections, and the βv_i connections correspond to self-inhibition. 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.

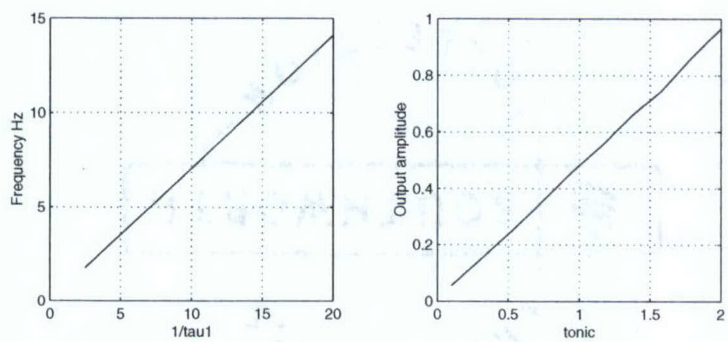

Figure 3-2: SIM Oscillator behavior under changing tonic c and time constant τ_1. The left hand figure shows the output frequency of the oscillator w_n plotted against $1/\tau_1$, and the right hand graph shows the amplitude of the oscillator output A_n plotted against the tonic excitation c. For this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

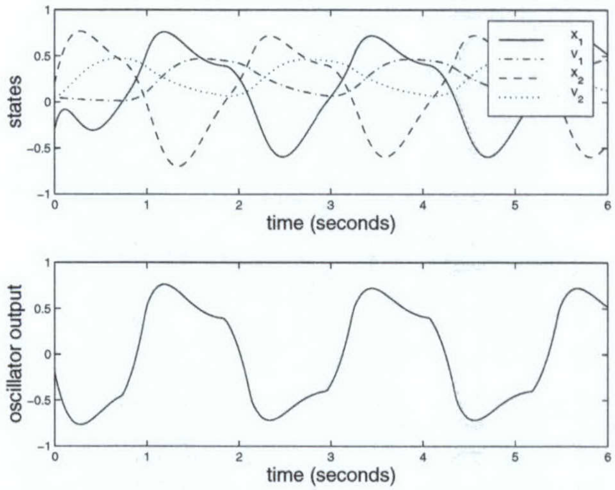

Figure 3-3: SIM A sample output from the oscillator. The top graph shows the variation with time of the states of the oscillator x_1, x_2, v_1, v_2 during normal operation. The bottom graph shows the output of the oscillator $y_{out} = [x_1]^+ - [x_2]^+$. For this example $\tau_1 = 0.25, \tau_2 = 0.5,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

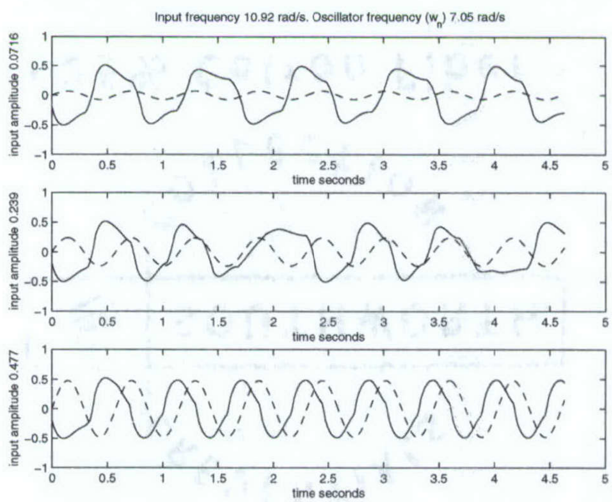

Figure 3-4: SIM Figure showing effect of increasing input signal. For small input (top graph), the oscillator is not entrained, and oscillates at its endogenous frequency w_n. In the middle graph, the input is larger, and the oscillator is almost entrained, but slips every couple of cycles. The lower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

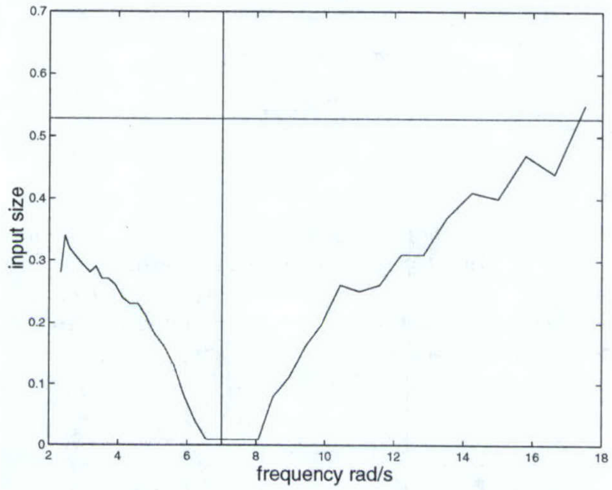

Figure 3-5: SIM Minimum input for entrainment. The figure shows the minimum input signal required for entrainment of the oscillator. The oscillator natural amplitude and frequency are given by the horizontal and vertical lines respectively. For this example $c = 1.0, \tau_1 = 0.1, \tau_2 = 0.2$.

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).

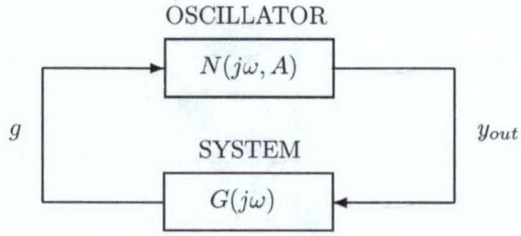

Figure 3-6: System schematic. The figure shows an oscillator tightly coupled to a system. The oscillator output y_{out} drives the input of the system, whose output g is the oscillator input. The analysis of the system proceeds by looking at the frequency responses of each system and calculating the 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

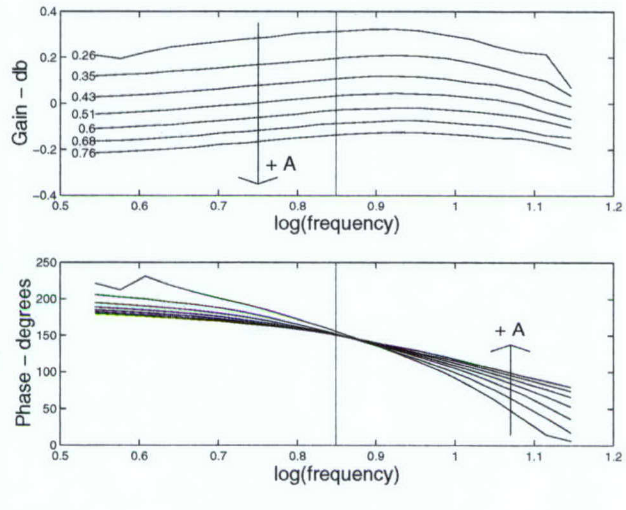

Figure 3-7: SIM Oscillator Bode plot. The top graph shows the gain $|N(j\omega, A)|$ plotted against frequency, and the lower plot the phase. The multiple lines correspond to different values of input amplitude A as indicated by the numbers and arrows on the plot. The gain is inversely related to A and 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.

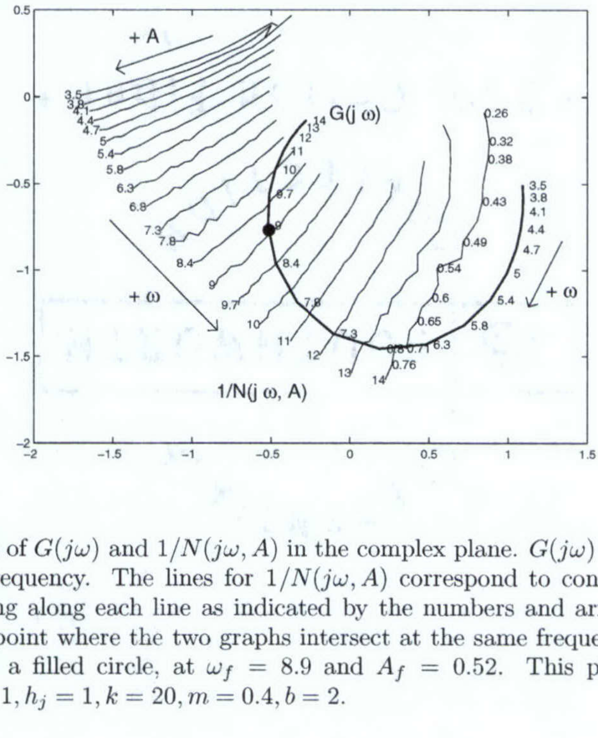

Figure 3-8: SIM Plot of $G(j\omega)$ and $1/N(j\omega, A)$ in the complex plane. $G(j\omega)$ is the thick line, with numbers indicating frequency. The lines for $1/N(j\omega, A)$ correspond to constant frequency, with amplitude A increasing along each line as indicated by the numbers and arrow. There 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 $\tau_1 = 0.1, \tau_2 = 0.2, c = 1, h_j = 1, k = 20, m = 0.4, b = 2$.

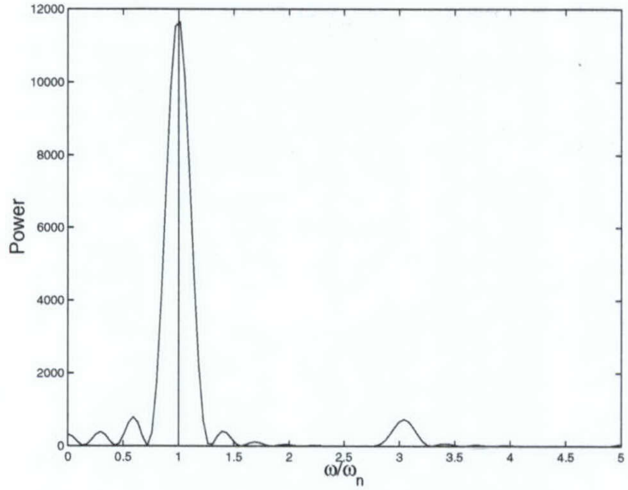

Figure 3-9: SIM Power density spectrum for the Matsuoka oscillator. The plot shows power plotted against frequency, expressed as a function of the oscillator natural frequency ω_n. The plot shows a clear peak at the natural frequency, and a much smaller one at $\omega = 3\omega_n$. The sinusoidal nature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

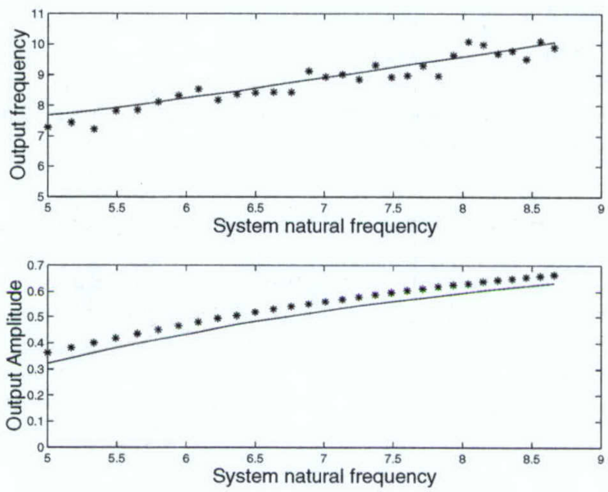

Figure 3-10: SIM Predictive accuracy. The plot shows the prediction (line) and measured (*) frequency (top plot) and amplitude (lower plot) for an oscillator driving a mass, as the stiffness of the spring was varied, so varying the natural frequency of the system. The accuracy is good, with a constant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

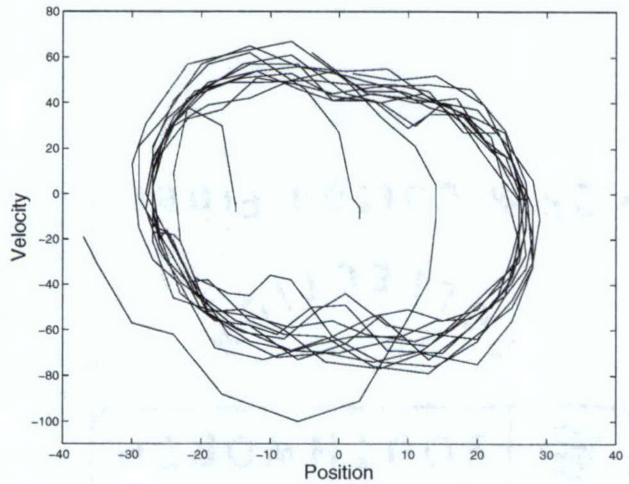

Figure 3-11: REAL Phase plot of robot motion. The plot shows velocity v. position for an oscillator driving a mass, recorded from one link of a robot arm. Shown are three traces overlaid starting from three different initial conditions. The three traces all converge to the limit cycle, indicative of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

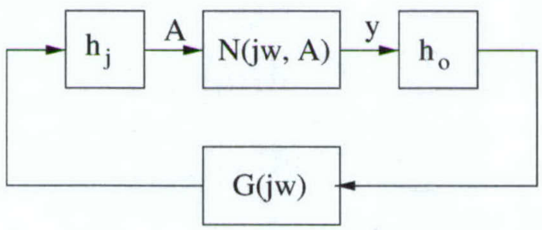

Figure 3-12: If N has an input gain h_j, and an output gain h_o, the overall gain of the oscillator becomes $h_j h_o N(j\omega, A)$. Since N depends on A, i.e. amplitude after the gain h_j, it makes sense to combine the gains with the plant transfer function G, i.e. $G'(j\omega) = h_j h_o G(j\omega)$.

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.

3.7 Effect of parameter changes (b): Robustness

The describing function analysis can also be used to examine the robustness of the oscillators. Robustness in this context means either that the coupled system has the same performance or behavior over a wide range of oscillator parameters, or that given a single set of parameters, a wide range of system properties give the same behavior. The oscillators are robust in both senses of the word. The oscillator parameters are generally well-behaved, with a smooth monotonic effect on the system behavior, decoupled from the effects of the other parameters.

Robust properties greatly ease the implementation of oscillator driven systems since very little tuning is required to achieve good results. This section evaluates each of the oscillator parameters in turn.

Input gains and tonics. The tonic parameter c affects the output amplitude of the oscillator and is effectively decoupled from the all the other parameters. The output amplitude is independent of input size and frequency (as shown in figure 3-7). This independence is a consequence of the non-linearities on the input g in equations (3.1) and (3.3). Appendix C shows that without these non-linear terms the system output is a function of input size, and is less well-behaved. The effect of the tonic parameter is monotonic, as was shown in figure 3-2.

The input parameter h is also decoupled from the oscillator parameters which set frequency. It has a more interesting effect on the overall system properties. In practice, the parameter does not have much effect on the final motion. It needs to be large enough to cause entrainment, but other than that its value is unimportant. The reason for this can be seen in figure 3-13, which plots the describing function for the oscillator with two different values of input gain. The two different solutions are $\omega_f = 8.82, A_f = 0.38$ for $h = 0.7$, and $\omega_f = 8.98, A_f = 0.77$, for $h = 1.7$. In spite of the gain more than doubling, the frequency changes by less than 2%. Since A_f is the amplitude of input to the oscillator, the actual motion has amplitude A_f/h, which is 0.55 when $h = 0.7$, and 0.45 when $h = 1.7$, a change of about 15%. Looking at figure 3-13, the reason for this robustness is the approximately radial lines of the oscillator describing function, particularly in the middle frequencies. The oscillator is inherently robust to changes in input size, giving approximately the same performance for a wide variety of parameter values.

For a fixed set of gains, the range of phases, frequencies, and gains that the oscillator can produce is large, so that any part of the system phase plot which overlaps with the area covered by the oscillator will produce an oscillatory solution. This contributes to the robustness of the oscillator itself.

Time constants. The main effect of changing the time constants of the oscillator is to change the frequency range of the oscillator. This effect is monotonic, as shown clearly in figure 3-2. As mentioned above, the effect is decoupled from the input gain and tonic parameters.

Referring to figure 3-13, if the lines were radial, then the effect of changing the time constants would be a monotonic change in the frequency of the coupled system. The lines are not exactly radial, which gives rise to the resonance tuning behavior of the system. The oscillator can tune into the resonant frequency of a driven system for a wide range of oscillator parameters, and conversely, for a fixed set of oscillator parameters, can tune into the frequency of a range of systems.

Figure 3-14 shows the range of settings of input gain and time constant which have a final solution which is within 10% of the system natural frequency. The plot shows that the oscillator natural frequency can vary from 3.5–5.5 rad/s which is a percentage change of 40 %, and the gain can vary even more (from 0.2 to 2), while still finding the resonant frequency. This was for a simulated system with $k = 20, m = 0.4, b = 5.2$.

Taking parameters for the oscillator in the middle of these values, i.e. $w_n = 4.5, h = 1$, the robustness of the oscillator to changing systems can be measured. The stiffness can be varied from 14–25.4 Nm/rad with the final frequency remaining within 10% of the natural frequency. This is a

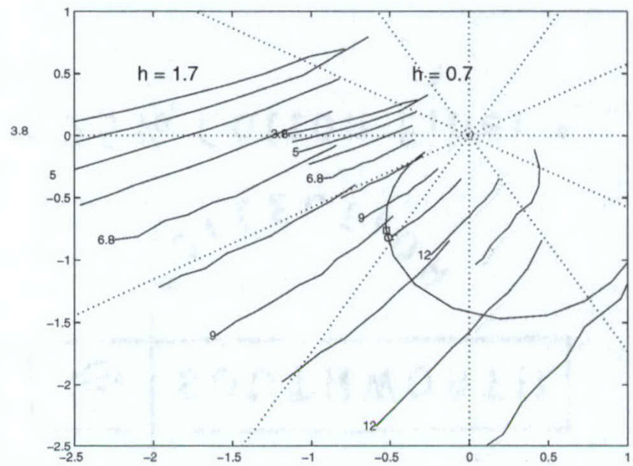

Figure 3-13: SIM Figure showing the effect of the gain on the describing function of the oscillator. The plot shows $1/N(j\omega, A)$ plotted on the complex plane for two values of input gain h, as marked. It also shows $G(j\omega)$ for a typical mass-spring system, with the limit cycles marked with a $\circ$ for $h = 0.7$, and a $\Box$ for $h = 1.7$. Because the lines of $1/N$ are approximately radial (especially for the middle frequencies), the effect of doubling the gain on the coupled system behavior is small. For high and low frequencies, changing the gain will alter how the oscillator couples with another system but the effect of the gain is still minor, given the acute angles between the lines and the radial directions.

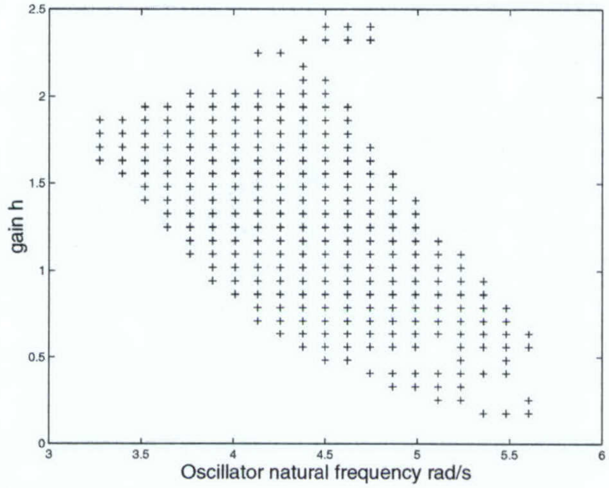

Figure 3-14: SIM Figure showing the range of values of the oscillator natural frequency, and oscillator input gain over which the final frequency of a coupled oscillator, mass-spring system was within 10% of the system natural frequency. The mass-spring system had values $k = 20, m = 0.4, b = 5.2$ corresponding to a natural frequency of 8.66 rad/s and damping ratio of 0.3.

30% change in stiffness, or approximately a 12% change in system resonant frequency. These results demonstrate clearly the robustness of the oscillator to parameter values and to changes in the system properties.

This resonance tuning behavior is the natural behavior of the coupled system. Since the oscillators entrain quickly, this means that the oscillator system is reactive to fast changes in the dynamics. An alternative method would be to identify the system and use a constant frequency command to excite the resonant frequency. While this method would probably give more accurate resonant tuning, it would not be responsive to changes in the dynamics without re-identifying the system.

The resonant frequency is also the most efficient speed to move the mass-spring system, with the minimum of real work required to produce the motion. The fact that the oscillator can find and drive at this frequency, while still adapting to the system properties is thus a useful behavior.

System changes The oscillator is also robust to other changes in the system properties. For example, changing the damping in the system alters the shape of $G(j\omega)$, but does not alter the frequency. Since the oscillator can produce a variety of gains and frequencies, it is automatically robust to these changes. For example, the damping ratio of the example system described above can be altered from 0.12 to 0.5 while still remaining entrained, with the entrained frequency varying from 7.56 to 6.59 rad/s where the resonant frequency is 7.07 rad/s, as shown in figure 3-15. This corresponds to a 7% change in frequency.

Entrainment The final aspect of the oscillator control which is practically useful is that it tends to entrain very quickly, usually within one cycle. This means that it is easy to tell whether a parameter is correct or not as the system settles quickly. This is also improves the robustness, since the oscillator responds quickly to changes in the dynamics.

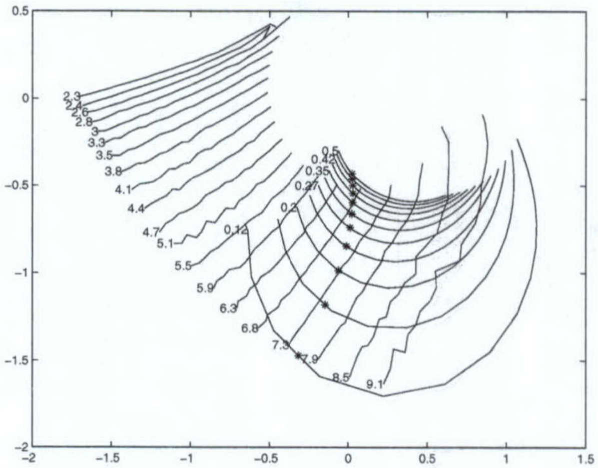

Figure 3-15: SIM Figure showing the effect of varying the damping ratio of the mass-spring system. The plot shows the describing function for the oscillator with lines for $G(j\omega)$ for different values of the damping ratio as marked. The damping ratio can be altered from 0.12 to 0.5 while remaining entrained, with only a 7% change in the output frequency away from the resonant frequency: frequency varies 7.56–6.69 rad/s, with the resonant frequency of 7.07 rad/s.

3.8 Design example: Resonance tuning

The describing function analysis is not only appropriate for analyzing the Matsuoka oscillator. This section shows how it can be used to analyze the well known Van der Pol oscillator, for the task of resonance tuning. There have been a number of papers concerning the resonance tuning property of neural oscillators (e.g. Hatsopoulos et al. (1992), Hatsopoulos and Warren (1996), Epstein and Kopell (1999)), without any full explanation of why this behavior is observed or predictions of its limits. The describing function analysis provides answers to both of those questions.

The Van der Pol (VDP) oscillator (Strogatz, 1994, p. 198), is described by the following equations:

$$\tau\dot{u} = wv + f(u) + h_j g_j \tag{3.13}$$
$$\tau\dot{v} = -\epsilon u \tag{3.14}$$
$$f(u) = u - u^3/3 \tag{3.15}$$
$$y_{out} = h_o u \tag{3.16}$$

where u and v are state variables, g_j is an input, h_j a gain on the input, h_o is a gain on the output, τ determines the speed of the oscillator, and ϵ is a parameter.[4]

The describing function for this oscillator is illustrated in 3-16, which shows a rather different pattern from the Matsuoka oscillator. The same trends are evident; the gain reduces with increasing A, although the output amplitude alters as a function of A, making the range of gains less uniform. The gain as before is roughly constant with frequency. The phase is roughly independent of amplitude, in the range $\pm 30°$.

[4]Reasonable values are $0.1 \leq \epsilon \leq 0.1, 1 \leq h_j \leq 30, 0.1 \leq h_o \leq 1$.

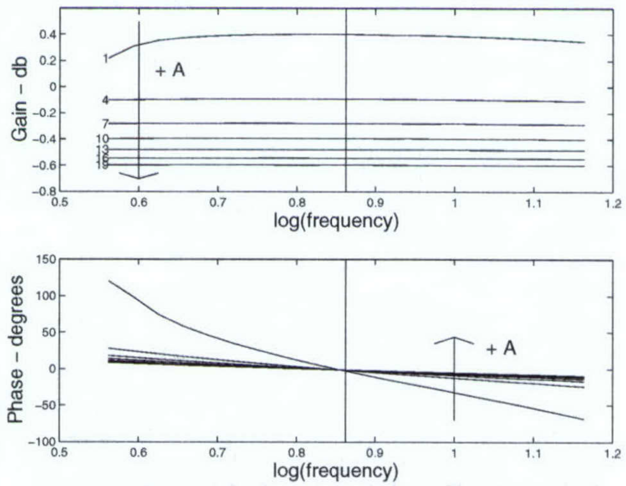

Figure 3-16: SIM Van der Pol Bode plot. The top graph shows the gain $|N(j\omega, A)|$ plotted against frequency, and the lower plot the phase. The multiple lines correspond to different values of input amplitude A as indicated by the numbers and arrows on the plot. The gain is inversely related to A and is roughly constant with frequency. Unlike the Matsuoka oscillator, the output amplitude of the VDP oscillator is a function of the input amplitude. This results in the lines in the gain plot not being equally spaced (compare to figure 3-7). The phase is less dependent on A, and is grouped around zero degrees. The natural frequency of the oscillator is indicated by the vertical line.

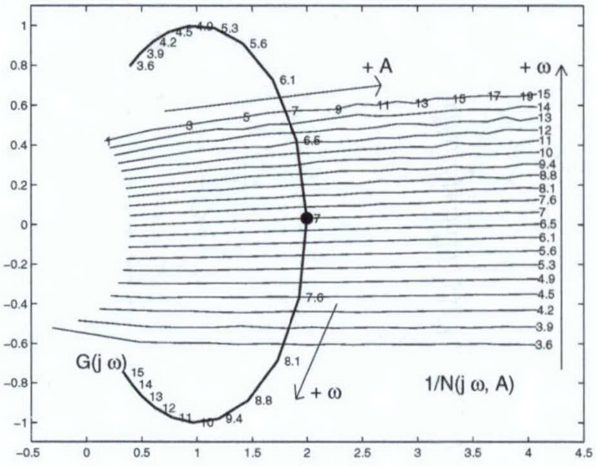

Figure 3-17: SIM Plot of $G(j\omega)$ and $1/N(j\omega, A)$ in the complex plane for the VDP oscillator. $G(j\omega)$ is the thick line, with numbers indicating frequency. The lines for $1/N(j\omega, A)$ correspond to constant frequency, with amplitude A increasing along each line as indicated by the 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 $\omega = 7.0$ and $A = 7.4$. This plot was produced for $\tau_1 = 0.03, \epsilon = 0.1, h_o = 0.2, h_j = 1, k = 20, m = 0.4, b = 2$.

If this oscillator is connected to a mass-spring system as before, because the Van der Pol oscillator can only produce phases in the range $\pm 30°$, the intersections would all be at low frequencies. However, if the velocity rather than the position is used as input, those phases correspond to points near the resonant frequency of the system. This is illustrated in figure 3-17, which shows $G(j\omega)$ for a mass-spring system with input θ_v, and output velocity $\dot{\theta}$, plotted together with $1/N(j\omega, A)$. The input and output gains are calculated to position the two plots over one another, here $h_j = 1, h_o = 0.2$. As before, there are limit cycle solutions where the graphs intersect at the same frequency, here $\omega = 7.0, A = 7.4$.

Because of the shape of the phase plot of the VDP, the oscillator can tune into the resonant frequency of the plant motion over a range of system frequencies. The Matsuoka oscillator also has this effect, but because the lines of its describing function are not as parallel, or as aligned with the resonant phase difference as the VDP characteristic, the resonance tuning behavior is not as strong. Using the describing function analysis, the resonance tuning effect of the VDP can be calculated, and measured in simulation. Figure 3-18 shows the result of this prediction. The plot shows the oscillator tuning into the resonant frequency of the system over a large range of frequencies. The accuracy of the describing function prediction is also shown in the graph.

To confirm this result, the VDP oscillator was implemented on the robot, and a similar experiment was conducted, varying the stiffness of a robot joint, and measuring the final frequency. The results from this are plotted in figure 3-19. The results show that over a range of frequencies corresponding to a four-fold increase in stiffness, the VDP oscillator drove the system at its resonant frequency.

This example illustrated not only the performance of the VDP oscillator system in this task, but also the use of describing function analysis to design and predict final system performance.

The VDP oscillator is in general less robust than the Matsuoka oscillator. This is because the

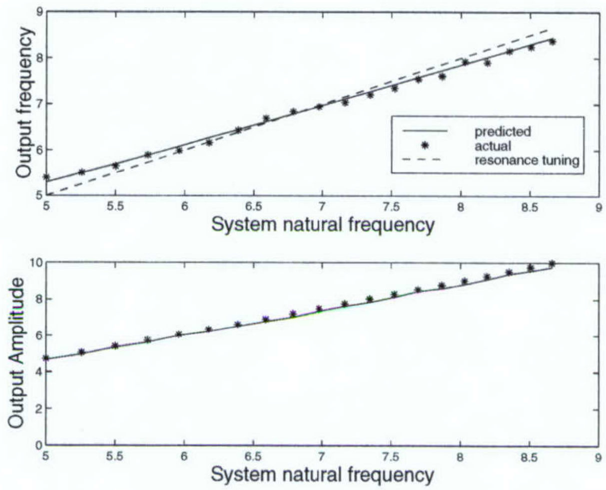

Figure 3-18: SIM Prediction of VDP oscillator driving a mass, using velocity as the oscillator input. The upper plot shows the accuracy of the prediction, and the resonance tuning behavior of the oscillator. The oscillator drives the mass at the resonant frequency of the mass-spring system, over a range of different conditions (here the frequency of the mass-spring system is varying from 5 to 9 rad/s). The lower plot shows the accuracy of the prediction of motion amplitude. For these plots $\tau_1 = 0.03, \epsilon = 0.1, h_o = 0.1, h_j = 2, k = 20, m = 0.4, b = 2$.

output amplitude is dependent on the input amplitude, making tuning difficult, and because the range of output phases is much less.

3.9 Design example: Juggling

This section describes the application of describing function analysis to a more complex task, that of juggling a ball on a paddle. The setup is illustrated in figure 3-20. This task has been addressed by a number of researchers, (Schaal and Atkeson, 1993, Rizzi and Koditschek, 1994), including one using neural oscillators (Miyakoshi et al., 1994). Using the method presented in this chapter, not only can the parameters be easily tuned without intensive simulation, but the robustness of the system can be calculated and compared to other approaches.

Schaal and Atkeson (1993) showed that when juggling a ball on a paddle, stable motion is achieved when there is a particular constant phase difference between the paddle and ball motion. This motivates the approach of writing the juggling problem as a describing function, where stable juggling is defined in terms of a phase between input (paddle motion) and output (ball motion), as well as a gain (the amplitude of the ball). Although this is a rough approximation to juggling, it allows analysis of the situation. The setup is shown in figure 3-21, where the paddle is assumed to move sinusoidally, i.e. $y(t) = B\sin(\omega t)$, and the ball bounces with height x. Since the juggling is stable, the velocity of the ball before and after impact is the same, so the impact equation is

$$(\dot{x} - \dot{y}) = -\alpha(-\dot{x} - \dot{y}) \tag{3.17}$$

where α is the coefficient of restitution. Calculating the time between bounces (which is $2\pi/\omega$), and

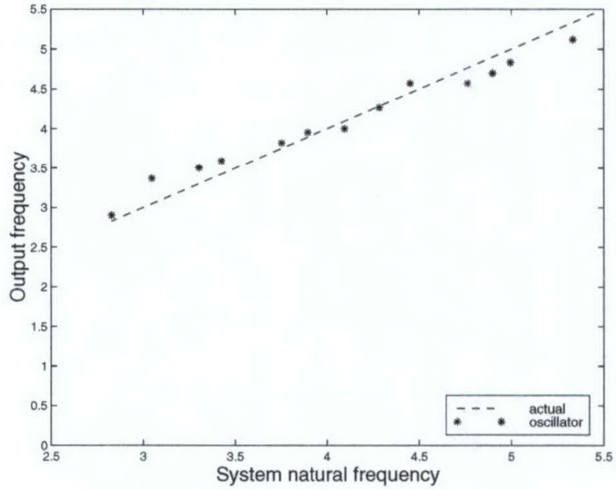

Figure 3-19: REAL Plot shows resonance tuning action of a VDP oscillator actuating a joint on the real robot. The stars mark measured frequency, plotted against the natural frequency of the system (also measured), as the stiffness of the system was altered. The results fit with the predictions of the describing function analysis. Because the natural frequency is a function of the square root of the system stiffness, the actual stiffness varied from 9 to 42.

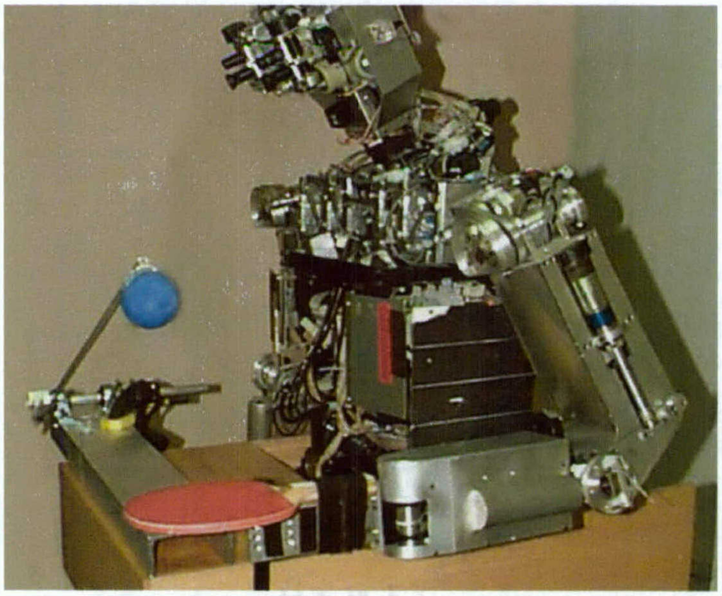

Figure 3-20: Picture of robot in juggling task. The paddle is a table tennis bat attached to the robot's arm. The ball is restrained to travel in one dimension by being mounted on a rotating boom. The angle of the boom is measured using a potentiometer at the pivot point, and this feedback signal is used by the oscillator to coordinate the arm motion with the ball motion.

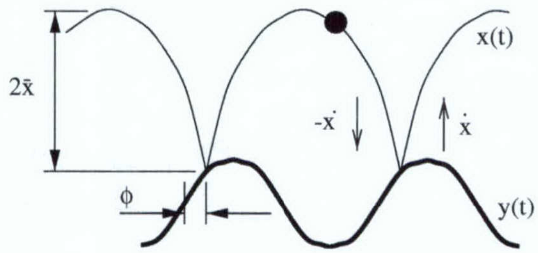

Figure 3-21: Schematic of the juggling action. The thick line shows the motion of the paddle $y(t) = B\sin(\omega t)$, and the thin line the ball trajectory $x(t)$. The ball impacts the paddle at a phase ϕ of the sine wave. The analysis expresses the juggling as a describing function, writing a stable solution in terms of the phase difference between ball and paddle trajectories $(\phi + \pi/2)$, and the amplitude of the ball motion $\overline{x}$.

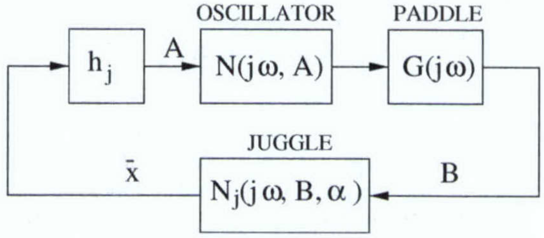

Figure 3-22: Oscillator driven juggling. The amplitude of the ball motion is $\overline{x}$, the input to the oscillator is $A = h_j\overline{x}$, and the output of the oscillator drives the paddle $(G(j\omega))$ with amplitude B. The ball motion can then be calculated using N_j. As before, there can be only a limit cycle when the loop gain is unity.

substituting $\dot{y} = B\omega\cos(\phi)$ at impact gives an equation for the phase at impact:

$$\phi = \arccos\left[\frac{\pi g}{B\omega^2}\left(\frac{1-\alpha}{1+\alpha}\right)\right] \tag{3.18}$$

Calculating the maximum height of the ball (at the time halfway between bounces) gives an expression for the ball amplitude:

$$\overline{x} = g\pi^2/4\omega^2 \tag{3.19}$$

This makes the describing function for the juggling $N_j(j\omega, B, \alpha) = \overline{x}\exp(-j(\phi + \pi/2))$. The paddle is driven by an oscillator with the same control system as before, i.e. through a spring, so that the oscillator output drives the desired position of the paddle, and the input is the ball trajectory x, as illustrated in figure 3-22. Since the juggle describing function is a non-linear function of B, which is itself generated by the non-linear oscillator/paddle, the functions cannot be separated and plotted as before. A loop gain, and a condition for stable oscillator driven juggling can still be calculated:

$$\overline{x} = N_j(j\omega, [h_j\overline{x}N(j\omega, h_j\overline{x})G(j\omega)], \alpha) \tag{3.20}$$

This can be solved numerically for the values of A, and ω which give stable juggling given the other parameters $(h_j, A_n, k, m$ etc.).

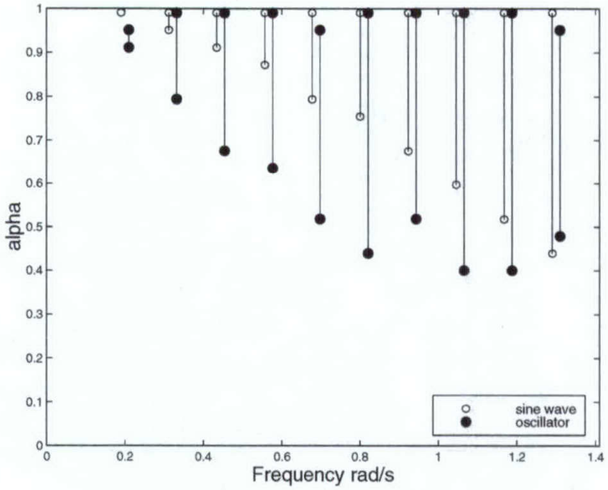

Figure 3-23: SIM Robustness to changes in α. The plot shows ranges of α for which stable juggling can be achieved by either an oscillator with natural frequency ω_n, ($\bullet$), or a constant frequency sine wave $B\sin(\omega_n t)$, ($\circ$), as the frequency ω_n is varied. Because the oscillator can alter its frequency, it is more robust to changes in α than the constant frequency solution, and thus more suitable for the task.

Solving this equation is a powerful tuning tool. Using the equation, it was found that using ball velocity $\dot{x}$ rather than the ball trajectory x as input gave more robust solutions. This would have been very difficult to discover without this analysis.

Solving this equation is also powerful for measuring robustness and comparing solutions. The robustness of the oscillator solution can be measured by fixing the oscillator parameters and calculating whether solutions exist for different values of restitution coefficient α. This can be compared to a similar calculation for a constant sine wave $y(t) = B\sin(\omega t)$. For the sine wave, the limit on α is given by real solutions of equation (3.18). The maximum value of α is 1, and the minimum is a function of frequency:

$$\alpha_{min} = \min(0, (1 - B\omega^2/g\pi)/(1 + B\omega^2/g\pi)) \tag{3.21}$$

Figure 3-23 shows the comparison between oscillators with natural frequency ω_n against sine waves of frequency ω_n, over a range of different ω_n's. The oscillators use velocity as input. The lines on the graph show the range of α's over which the different methods could stably juggle. Because the oscillator can produce outputs over a range of frequencies, it can juggle with a wider range of α's than the constant frequency solution. This result shows that oscillators are a better solution than a constant sine wave for this task.

Motivated by these results, juggling was implemented using a paddle attached to the arm of the robot. The setup is shown in figure 3-20. The ball was restricted to move in one dimension by a lightweight boom whose angle was measured using a potentiometer. As indicated by the analysis, ball velocity was chosen as the input signal to the oscillator. Figure 3-24 shows a short transient from the juggling performance. The juggling is much more variable than a simulated version, since there is a great deal of noise not only in the sensor signal, but also in the way the robot hits the tethered ball, effectively altering α for each impact. The change in speed of the oscillator driven paddle is visible in the plot. The oscillator responds on a hit by hit basis, remaining coordinated

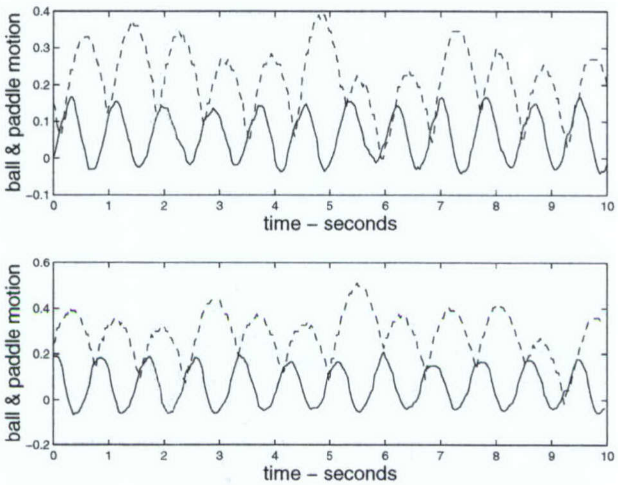

Figure 3-24: REAL Plot showing paddle position (solid) and ball trajectory (dashed) against time for 2 traces of the robot juggling. Because of the way the arm hits the ball, there is significant noise in the system. The oscillator responds to the changing amplitude of the ball by adjusting its speed slightly, which is noticeable in the middle of the trace.

with the ball motion.

3.10 Adaptation

As well as responding to the type of system, and adjusting its frequency with respect to the system properties, the oscillator synchronizes itself with the mass motion. This was seen in the juggling example, the oscillator coordinating the motion of the paddle with the height of the ball on a hit by hit basis. The same action occurs when driving a single robot joint. This is because the oscillator is tightly coupled, and the oscillator output is referenced relative to its input, which is the motion of the joint. This means that the oscillator is sensitive to changes in the motion of the joint. For example if the oscillator is moving a certain link back and forth, and something or someone moves the link, the oscillator entrains with the imposed motion, and remains synchronized. This is illustrated in figure 3-25. The oscillator is sensitive to the exact motion of the joints, and not just the general system properties. This behavior and sensitivity is important and useful for providing coordination between the joints through the natural dynamics. This topic is examined in more detail in chapter 4.

3.11 Conclusion

This chapter has introduced the oscillators, and described their behavior coupled to single degree of freedom mechanical systems. Analysis using describing functions was introduced, and was shown to be powerful for analyzing the motion, predicting the behavior of the coupled system, and a practical tool for tuning, evaluating the robustness of the parameters, and evaluating the oscillator approach.

Using this analysis on oscillators reveals some general principles which apply to the oscillator solutions.

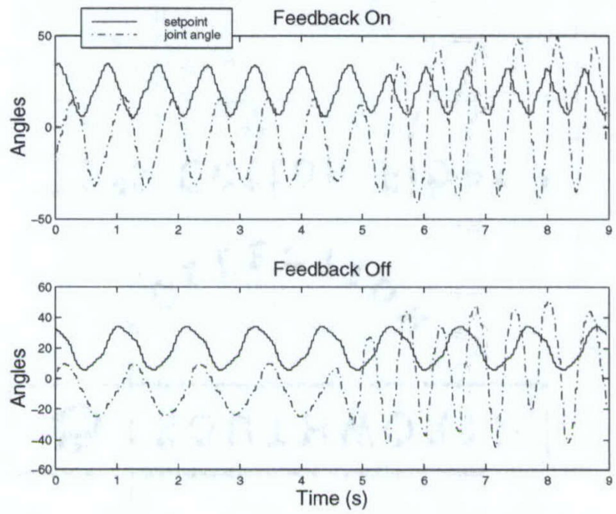

Figure 3-25: REAL Effect of external disturbances on the oscillator entrainment. The plots show the oscillator outputs (solid) and the joint angle (dash dot) for a single joint of the robot. Approximately halfway through the trace, the arm was moved by the author at a higher frequency and larger amplitude. When feedback is used (top graph), the oscillator responds to the extra motion, and remains entrained, its output tracking the imposed motion. In the lower trace when the feedback is off, the oscillator ignores the imposed motion, which is thus more jerky, since the oscillator and the disturbance are not coordinated.

- Robustness in parameter settings. Due to the wide range of gains that the oscillator can produce, the oscillator will entrain with a wide range of different systems. The behavior of the oscillator also makes the parameters of the oscillator robust because changing them either has a minor, or a predictable effect. The effect of varying parameters is smooth, monotonic, and they are decoupled from one another. This has the consequence that the system is easy to use in practice, it being easy to choose working parameters.

- Sensitivity to system changes. Again due to the shape of N, oscillator systems are sensitive to changes in the resonant frequencies of the mechanical systems, driving them at frequencies close to the resonant frequency. This occurs over a wide range of oscillator parameters, and a range of system frequencies. The oscillator behavior is automatic and is reactive, altering quickly when the system dynamics change. Since the resonant frequency is an efficient speed to move a mass-spring system, the oscillator behavior is finding the best speed, dependent on the natural dynamics.

- Stability of final solutions. Due to the shape of the plots, the oscillators produce stable motions actuating common mass-spring mechanical systems. This together with their observed fast entrainment to limit cycle solutions makes them well suited for practical applications.

- Synchronization. The oscillator produces a command to the joints which is defined relative to the joint motion. The oscillator is thus sensitive and responds to the joint motion, driving slower or faster when the joint is forced by the system, be it another joint of the arm or a human grabbing and moving the arm. This adaptation is crucial to the oscillator behavior when driving multiple degrees of freedom, as described in the following chapter.

Chapter 4

Coupling through natural dynamics

This chapter examines the behavior of the oscillators while driving multiple degrees of freedom of the arm, as opposed to the single degree of freedom motions examined in the previous chapter. The oscillators are connected to drive each joint of the arm independently, as shown in figure 4-1. There is a tightly coupled oscillator at each joint, with no connections between them. The oscillators use mechanical coupling through the physical arm to coordinate with one another and the task.

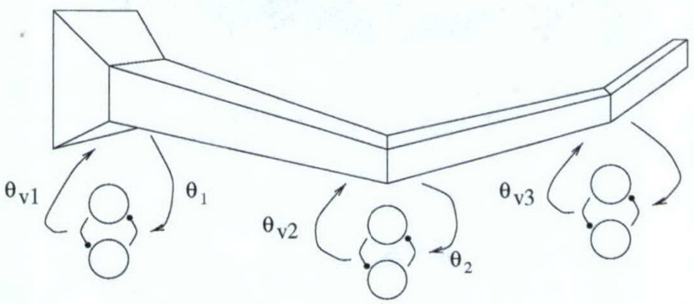

Figure 4-1: Figure showing configuration of oscillators and arm throughout this chapter. The oscillators are tightly coupled to each joint, with their output driving the setpoint θ_{vi}, and their input being either the joint angle θ_i, or the joint torque u_i. There are no software connections between the oscillators; they use mechanical coupling through the physical arm detected using the feedback to coordinate with one another and the task.

The versatility of this configuration is demonstrated using a number of examples, including crank turning using both one and two arms, and with the arms in both redundant and non-redundant configurations. Other position constrained tasks such as pumping a bicycle pump have also been demonstrated. The configuration can also exploit the dynamics of objects for coordination of the joints; the chapter includes the example of two arms coordinated through the dynamics of a Slinky toy, which is passed from hand to hand.

The analysis tools introduced in chapter 3 are extended to analyze these multi-degree of freedom motions. The analysis indicates again the sensitivity of the oscillators, showing that their behavior is to drive the resonant mode of the mechanical system. This is appropriate for a range of tasks, for example, attaching a crank to the arm creates a low frequency resonant mode of the system, which the oscillators automatically find and drive.

The crank turning example is analyzed in the context of these results, showing also the robustness of the oscillator solutions, with many different parameter settings giving crank turning, and a single set of oscillator parameters being robust to large changes in the arm dynamics.

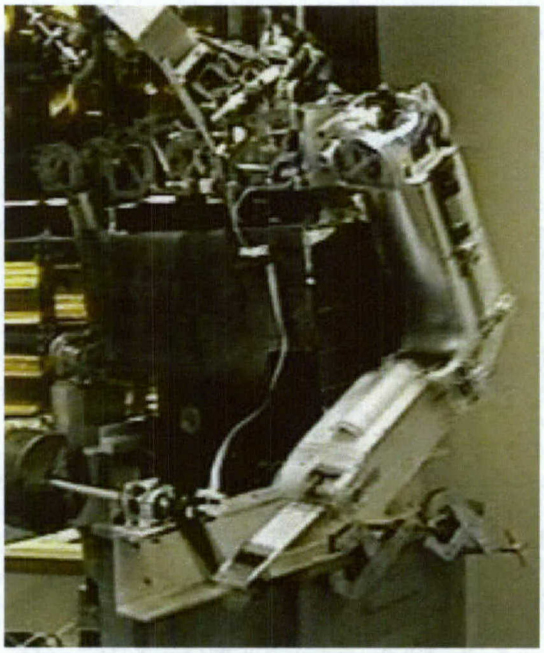

Figure 4-2: REAL Crank turning with one arm. The picture shows the robot using its left arm to turn the crank. Oscillators are used at the shoulder and elbow in this configuration. The oscillators are initially uncoordinated, but use feedback from the motion of the joints to adjust their outputs, so turning the crank.

The chapter as a whole points out the self-organizing property of the oscillators, performing complex mechanical tasks which require coordination between the joints in a simple and robust manner.

4.1 Examples

The first example is crank turning. The oscillators have been used to turn cranks in a variety of different configurations, the most simple of which is shown in figure 4-2. Here two oscillators driving the shoulder and elbow joints are used to turn the crank in the plane. They use feedback of the joint angle to modify their outputs. As described above, there is no explicit connection between the oscillators at the two joints, their only connection being through the mechanics of the arm, and their feedback.

When the feedback is off, the oscillators oscillate at their natural frequencies, producing uncoordinated rhythmic drives to the arm joints. When the feedback is switched on, the oscillators respond to the dynamics of the situation, become coordinated with one another, and turn the crank smoothly, as shown in the transients in figure 4-3. Because the oscillators entrain rapidly, the crank turning settles down rapidly to a steady motion. The crank turning behavior is robust to changes in most of the oscillator parameters (gains, time constants etc), and is sensitive more to the posture of the arm, and the sizes of the oscillator amplitudes (tonic). The posture and the amplitudes determine whether the arm goes all the way round, and whether the final motion is smooth and energy efficient, or more violent. Because the arm is compliant, the system is still fairly robust to these values, with a variety of postures and amplitudes giving crank turning motion. This is considered

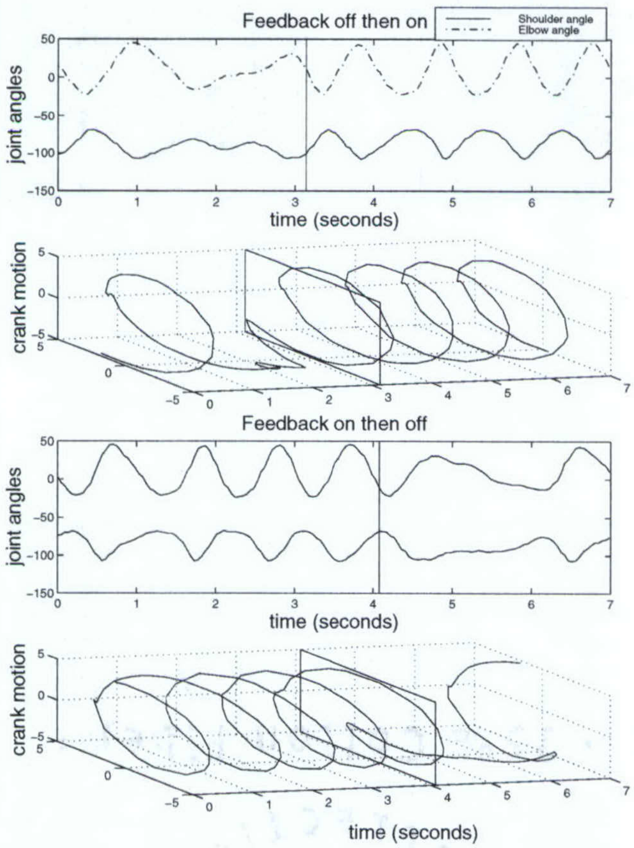

Figure 4-3: REAL Figure showing the effect of feedback for crank turning using two degrees of freedom in the configuration shown in figure 4-2. The lines show the angles of the joints. When the feedback is on, the oscillators are coordinated with one another, and the crank is turned (the crank angle is the dash-dot line, which wraps around at 180 degrees). Without feedback the oscillators revert to their natural frequencies, which does not result in smooth crank turning. The glitch in the crank trace is caused by the angle sensor wrapping round, not by a jerk in the motion.

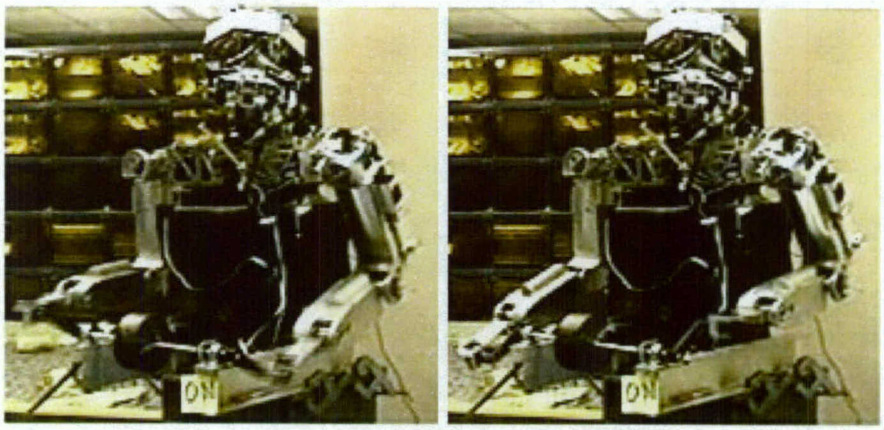

Figure 4-4: REAL Crank turning with more than one arm. Since the oscillators are independent, the crank turning can be easily extended to use more than one arm. Here two arms are used to turn the crank, using four oscillators with two on each arm. As before, the oscillators are synchronized through mechanical coupling, there being no explicit connections between the oscillators on a single arm, or between the two arms.

in detail in section 4.7.

Since the control of the joints is independent, it makes no difference whether one or two arms are used. Figure 4-4 shows two arms turning the crank where each arm is in the same configuration as before. Four oscillators are used to achieve the motion, driving both shoulders and both elbows.

The crank is connected to the arm, and thus provides a strong constraint on how the arm can move. This is particularly the case in the configurations shown in figures 4-2 and 4-4 because the arm is not in a redundant configuration.[1] The motion of the crank determines how the shoulder and elbow joints can move. Because the oscillators use feedback from the joint angle, and because they provide an output which is a certain phase difference from their input signal (as described in chapter 3), they respond to the constraint by driving the arm around it.

The oscillators can also find crank turning solutions when the arm is redundant, i.e. configurations where the crank does not completely specify how the joints should move. A redundant configuration is shown in figure 4-5, where between four and six degrees of freedom can be used to turn the crank. The turning in this redundant case has the same properties as the non-redundant configurations: coordination through mechanical coupling, quick entrainment (see figure 4-6), and robustness to parameter changes. The oscillators have also been used for other constrained tasks such as pumping a bicycle pump.

Since the oscillator outputs are approximately sinusoidal, and the joint angle motions induced by the constraint are not sinusoidal (as in figure 4-6), there is a considerable tracking error. If the arm were stiff, this would be a problem and might cause the arm to jam. However, because the arm is compliant, the error is absorbed in the arm compliance, and the final motion is smooth. The compliance of the arm thus gives robustness. In addition, the compliance together with the position constraint expands the repertoire of the oscillators. The system exploits the crank itself to create a motion which is not solely created by rhythmic commands at the joints.

There have been many robotic approaches to crank turning, for example using impedance control

[1] There are actually three joints in the plane of the arm, the shoulder, elbow and wrist, which makes this configuration redundant. However, the wrist joint was not actuated and maintained a constant stiffness, so to all intents and purposes the configuration is not redundant.

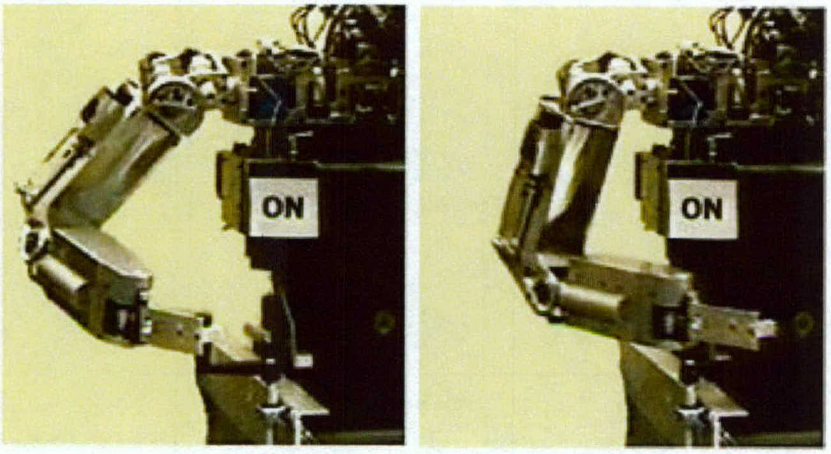

Figure 4-5: REAL Crank turning using a redundant arm. The oscillators can be used to turn cranks even when the arm is redundant, as shown here. The arm uses 4 oscillators, driving 4 degrees of freedom, two in the shoulder, and two in the elbow. The transients from this system are shown in figure 4-6.

(Hogan, 1985a), or hybrid force/position control (Raibert and Craig, 1981). These approaches use knowledge about the arm and crank kinematics to both move the arm to the crank, and to coordinate multiple degrees of freedom to turn the crank. They use this kinematic knowledge to deal with cranks of different sizes and different locations. The oscillator solution requires no calibration or kinematic transformations, but is only solving half of the problem. The arm starts connected to the crank, the posture and oscillator amplitudes set for the particular crank size and position, and the oscillators only provide the coordination for the motion. Even for creating the motion, the oscillator solution is more simple, with uncalibrated identical local controllers interacting through the arm mechanics, compared to a single complex calibrated kinematic controller.

The reason why the local control works is that the arm is compliant, making the interaction with the crank robust and giving some dynamics to exploit, and that the oscillators have the right properties to be local controllers for these tasks. These properties are examined later in this chapter.

Even in the redundant case, the crank provides a strong constraint on how the arm can move. However, the oscillators do not need such a strong constraint in order to coordinate multiple degrees of freedom through mechanical coupling. A different task accomplished in this manner is passing a Slinky toy from hand to hand, as illustrated in figure 4-7. An independent oscillator is used to control each arm, only coupled by mechanical interactions with the Slinky toy itself. The oscillators quickly converge on an out of phase motion of the Slinky, as shown in figure 4-8. The coupling forces from the Slinky are small compared to the crank forces, but are still large enough to entrain the oscillators.

In all these examples, the oscillators are using the mechanical coupling though a physical structure to become coordinated with one another. This allows them to perform complicated tasks in a very simple manner. In the following section a simple model of this coupling is developed. The model shows that the behavior of the oscillators is to find the resonant mode of the mechanical system. Connecting the arm to the crank, or connecting the arms with a Slinky can thus be seen as altering the natural dynamics of the system so that the resonant mode is the movement required to successfully complete the task.

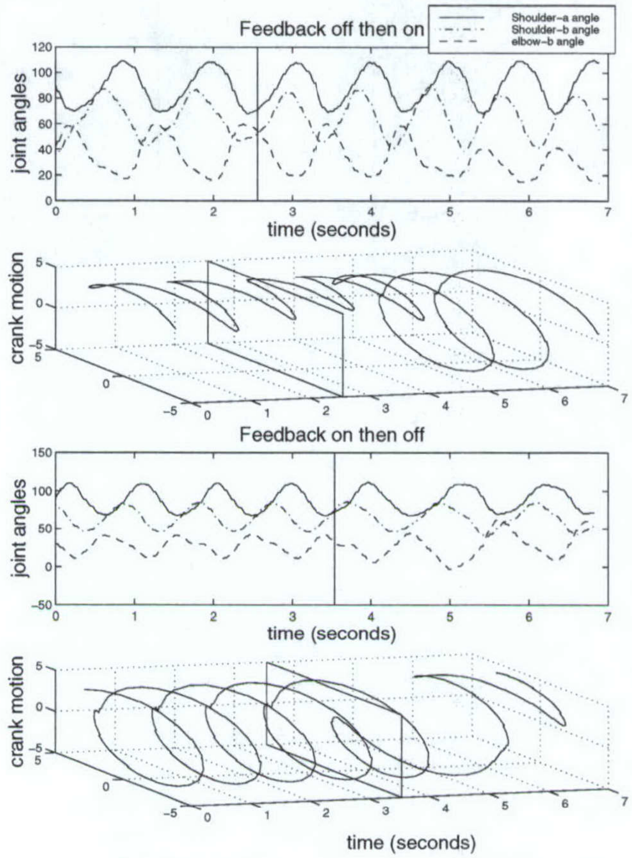

Figure 4-6: REAL Transients of crank turning with a redundant arm (the configuration shown in figure 4-5). Three degrees of freedom are shown here, for two shoulder joints and one elbow joint. The top two graphs show the joint angles, and the motion of the crank as the feedback is turned on, which occurs at the vertical line. The system has a short transient before finding the stable motion. The lower two plots show the same result, only this time the feedback is turned off at the vertical line. The system quickly falls out of coordination, and the crank stops being turned smoothly. The glitch in the crank trace is caused by the angle sensor wrapping round, not by a jerk in the motion.

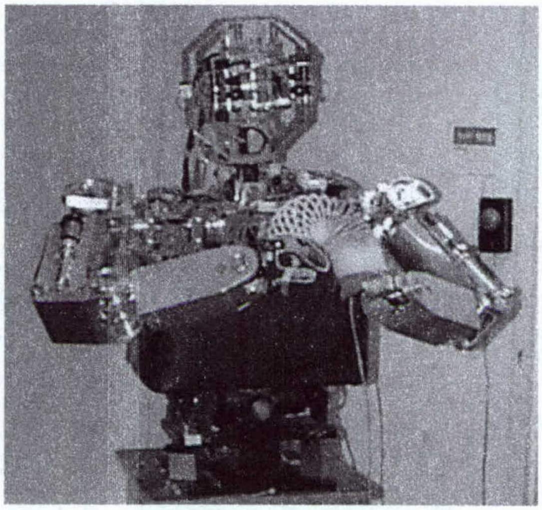

Figure 4-7: Picture of Cog passing the Slinky toy from hand to hand. The two elbow joints are used to move the hands up and down, where the coordination between the hands is given by the interaction between the oscillator dynamics and the coupled arm-slinky system.

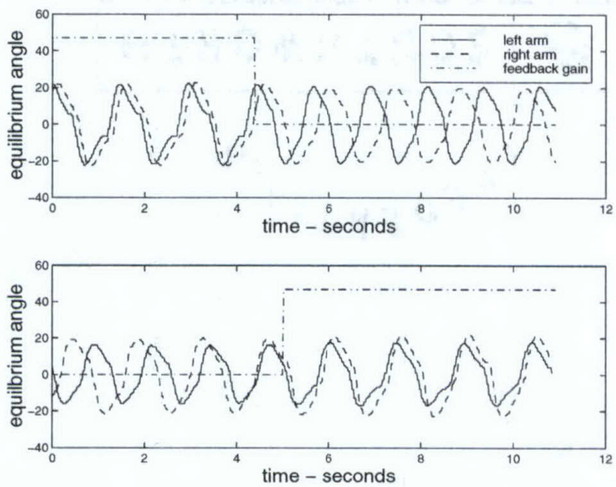

Figure 4-8: REAL Two examples of Slinky operation. Both plots show the outputs from the oscillators as the torque feedback (dash-dot) is turned on and off. When the traces are in phase, the Slinky is moving in anti-phase. When the feedback is on, the two arms are coordinated and the outputs are synchronized, but when off, the oscillators are no longer synchronized. The only connection between the oscillators is through the physical structure of the Slinky.

4.2 Coupling model

The action of the mechanical coupling through the arm is to constrain the motion of each joint dependent on the motion of the other joints. Since the arm is compliant and in most configurations redundant, the coupling is not very stiff, i.e. there is some slop in the system. A simple approximation to this coupling is shown in figure 4-9. This shows two robot links represented as masses, with the joint level control appearing as a spring and damper connected to each mass. The coupling is included as a spring coupling the two masses, the stiffness of which can be varied to model the strength of the coupling. Using describing function analysis to represent the oscillators driving each mass extends the results in chapter 3 to this higher dimensional system.

In detail, the coupling model in figure 4-9 consists of two masses m_1, m_2 driven through springs k_1, k_2 and dampers c_1, c_2 by oscillators. The motion of each mass is coupled to the other by the coupling spring k_T.[2] The motion or angle of each mass is θ_1, θ_2, and the oscillator outputs are θ_{v1}, θ_{v2}.

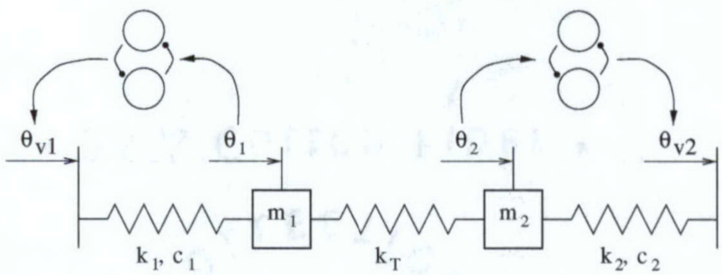

Figure 4-9: A simple model of coupling through the natural dynamics. The model consists of two masses driven by oscillators, connected by a coupling spring k_T.

The equations of motion for this system are

$$\begin{aligned} m_1\ddot{\theta}_1 + c_1\dot{\theta}_1 + (k_1 + k_T)\theta_1 - k_T\theta_2 &= k_1\theta_{v1} \\ m_2\ddot{\theta}_2 + c_2\dot{\theta}_2 - k_T\theta_1 + (k_2 + k_T)\theta_2 &= k_2\theta_{v2} \end{aligned} \quad (4.1)$$

This is a resonant mass-spring system, which has a free-vibration behavior when there is no driving input (i.e. $\theta_{v1} = \theta_{v2} = 0$). There are two resonant modes, at two different resonant frequencies. These can be determined by assuming solutions of the form $\Theta = Ae^{j\omega t}$, and solving the eigenvalue problem

$$\begin{pmatrix} (k_1 + k_T)/m_1 & -k_T/m_1 \\ -k_T/m_2 & (k_2 + k_T)/m_2 \end{pmatrix} \begin{pmatrix} \Theta_1 \\ \Theta_2 \end{pmatrix} = \omega^2 \begin{pmatrix} \Theta_1 \\ \Theta_2 \end{pmatrix} \quad (4.2)$$

for values of ω and ratios Θ_1/Θ_2.

When this system is driven by an oscillator, the behavior of the oscillator can be modeled using the describing function analysis developed in chapter 3. The oscillator behavior is expressed as a transfer function $N_i(j\omega, A)$, with a gain g_i and phase ϕ_i, evaluated over a range of frequencies and input amplitudes, i.e.

$$g_i e^{j\phi_i} = N_i(j\omega, A) \quad (4.3)$$

[2] In general the coupling should include a damper. Including the damper complicates the equations without changing their fundamental result so it was ignored.

Because the oscillator provides a driving force which is a function of the joint angles Θ, the effect of the oscillator is to turn the system from being a driven resonant system (4.1), to a freely vibrating system:

$$\begin{pmatrix} (k_1 + k_T + j\omega c_1 - k_1 g_1 e^{j\phi_1})/m_1 & -k_T/m_1 \\ -k_T/m_2 & (k_2 + k_T + j\omega c_2 - k_2 g_2 e^{j\phi_2})/m_2 \end{pmatrix} \begin{pmatrix} \Theta_1 \\ \Theta_2 \end{pmatrix} = \omega^2 \begin{pmatrix} \Theta_1 \\ \Theta_2 \end{pmatrix} \tag{4.4}$$

If this system has a steady state vibrating solution, then the solutions for ω must be real, since complex values of ω correspond to solutions which either decay or grow. This implies that the oscillator must cancel out the damping in the system, or in this case that the imaginary parts on the diagonals must be zero (Strang, 1993).

$$\omega c_1 - k_1 \mathrm{Im}[g_1 e^{\phi_1}] = \omega c_2 - k_2 \mathrm{Im}[g_2 e^{\phi_2}] = 0 \tag{4.5}$$

If the two oscillator driven systems are assumed to have the same damping factor ζ then this equation can be simplified. Writing $\omega_{ni} = \sqrt{k_i/m_i}$ and $2\zeta\omega_{ni} = c_i/m_i$ for each mass, the equation becomes

$$2\zeta\omega = \omega_{n1} g_1 \sin\phi_1 = \omega_{n2} g_2 \sin\phi_2 \tag{4.6}$$

This defines an important relation between the two oscillators, as well as a relation between the gain and phase of the individual oscillators and frequency. Because g_i and ϕ_i are both functions of frequency this equation can be solved numerically, fixing the frequency, and finding the value of A where

$$2\zeta\omega = \omega_{ni} \mathrm{Im}[N_i(j\omega, A)] \tag{4.7}$$

Figure 4-10 shows the solution to this equation for a typical choice of values for the masses and springs. The solution for each oscillator is indicated by the squares and stars on the plot. There are a range of solutions which vary in frequency, the extent of the solutions indicating the range of frequencies over which the oscillators can cancel out the damping, and produce steady state solutions.

The actual frequency of the motion depends on the solution for the eigenvectors of the complete system. When the imaginary parts have been removed, the problem reduces to

$$\begin{pmatrix} (k_1 + k_T - k_1 \mathrm{Re}[g_1 e^{j\phi_1}])/m_1 & -k_T/m_1 \\ -k_T/m_2 & (k_2 + k_T - k_2 \mathrm{Re}[g_2 e^{j\phi_2}])/m_2 \end{pmatrix} \begin{pmatrix} \Theta_1 \\ \Theta_2 \end{pmatrix} = \omega^2 \begin{pmatrix} \Theta_1 \\ \Theta_2 \end{pmatrix} \tag{4.8}$$

The effect of the oscillator is through the real part of its transfer function, $\mathrm{Re}[g_i e^{j\phi_i}] = g_i \cos\phi_i$. If the phase $\phi_i = 90°$, then this term is zero, and the equations reduce to the underlying mechanical system. This would imply that the oscillator driving this system simply removed the damping, but did not interfere with the shape of the motion. Unfortunately the phases are not exactly 90°, as show in figure 4-10. They lie in the range $90° - 160°$, which means that they affect the system. However, as the analysis below will show, they affect the frequency of the oscillation far more than the final mode shape.

The expression for the eigenfrequencies is obtained by solving (4.8) and is complicated:

$$\omega^2 = \frac{1}{2}\left[G_1 + G_2 + k_T(\frac{1}{m_1} + \frac{1}{m_2}) \pm \sqrt{(G_1 - G_2)^2 + 2k_T(\frac{1}{m_1} - \frac{1}{m_2})(G_1 - G_2) + k_T^2(\frac{1}{m_1} + \frac{1}{m_2})^2}\right] \tag{4.9}$$

where $G_i = (k_i/m_i)(1 - g_i \cos\phi_i)$. The values of g_i and ϕ_i are taken from the solutions to the damping constraint and are both functions of frequency. The equation can be solved numerically, with the

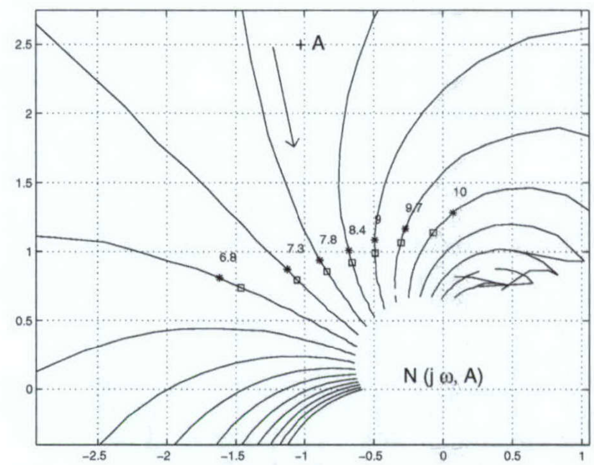

Figure 4-10: SIM Figure showing solutions to the damping constraint (4.6). The plot shows $N(j\omega, A)$ in the complex plane, where the lines correspond to constant frequency, with increasing amplitude A as shown by the arrow. The squares and stars show the points where the damping in the system is zero ($*$ - mass 1, $\square$ - mass 2). The numbers show the frequencies of these solutions. The range of solutions indicates the range of frequencies over which the oscillators can drive the system. The solutions at the same frequency are close together, with similar amplitudes and phases. The phases vary from about 90° for $\omega = 10$ to near to 160° for $\omega = 6.8$. For this example, $k_1 = 30, m_1 = 1, c_1 = 4.65, k_2 = 25, m_2 = 1, c_2 = 3, k_T = 30$.

graphical interpretation of the solution shown in figure 4-11. This shows ω^2 plotted together with the right hand side of (4.9) evaluated at the damping solutions. In this case there are two possible solutions where the lines intersect, at $\omega = 7.4, 9.7$. The two solutions correspond to two different modes of the system motion, which can be calculated by evaluating the eigenvectors at those solution frequencies.

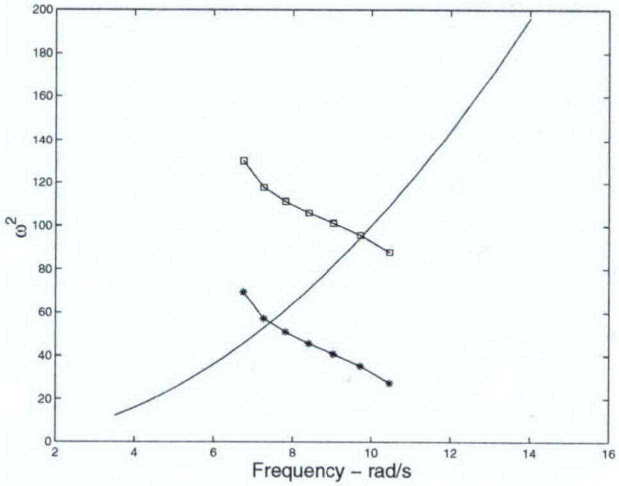

Figure 4-11: SIM Figure showing solutions for the final system frequency. The plot shows ω^2 plotted together with the expression for the eigenvalues (equation (4.9)) evaluated at the oscillator solutions to the damping constraint. There are steady state solutions where the lines intersect, in this case there are two solutions at $\omega = 7.4, 9.7$. The plot shows that the oscillators produce at most as many solutions as there are modes of the underlying system, since there are only two possible intersections here.

The expression for the eigenvectors (4.9) carries little intuition, but can be simplified by considering the relationship between the damping constraint and the oscillator properties. The damping constraint (4.6) implies that the imaginary part of N multiplied by the local resonant frequency ω_{ni} is constant. The oscillators behavior is such that at constant frequency, the phase of N does not change greatly with input amplitude (see chapter 3, figure 3-7). This is also evident in figure 4-10, where the phases for the two solutions are approximately equal. Under this assumption,

$$\begin{aligned} G_1 - G_2 &= (\omega_{n1}^2 - \omega_{n2}^2) - (\omega_{n1}^2 g_1 \cos\phi_1 - \omega_{n2}^2 g_2 \cos\phi_2) \\ &\approx (\omega_{n1}^2 - \omega_{n2}^2) \end{aligned} \tag{4.10}$$

Assuming that this can be neglected compared to the other terms in the eigenvalue expression, (4.9) can be simplified, giving the two values of frequency ω:

$$\omega^2 \approx \begin{cases} (G_1 + G_2)/2 + k_T(\frac{1}{m_1} + \frac{1}{m_2}) \\ (G_1 + G_2)/2 \end{cases} \tag{4.11}$$

From which the eigenmodes of the system can be calculated. The ratio Θ_1/Θ_2 also has two values:

$$\Theta_1/\Theta_2 \approx \begin{cases} \frac{2k_T/m_1}{k_1/m_1 - k_2/m_2 - 2k_T/m_1} \\ \frac{2k_T/m_1}{k_1/m_1 - k_2/m_2 + 2k_T/m_2} \end{cases} \tag{4.12}$$

which is approximately the same as the resonant modes of the underlying mechanical system. Figure 4-12 shows the comparison between the modes of the mechanical system, and the modes found by the oscillator in this case.

Interestingly and importantly, the oscillator only finds one mode of the system. This is unlike the linear case, where the final solution is normally a superposition of the eigenmodes. One important constraint on the oscillator behavior is that it can only entrain over a limited range of frequencies. If the mode of the original system has a natural frequency which does not lie in this range, then the oscillator cannot have that mode as a periodic solution i.e. the lines in figure 4-11 will not intersect.

There are cases where there are two possible stable solutions, as in the example above. In that case the oscillator finds one or the other depending on initial conditions. Figure 4-14 shows two time traces of the transients finding either of the two modes for the example above. The reason for finding one solution is that the two modes correspond to different entrained frequencies for the oscillator. The oscillator can only output one frequency, (see chapter 3, figure 3-9), and thus cannot drive both modes simultaneously. This characteristic appears to be peculiar to the Matsuoka oscillator, as the Van der Pol oscillator (introduced in section 3.8) can drive two modes at once, as shown in figure 4-15. Driving two modes is undesirable for most applications.

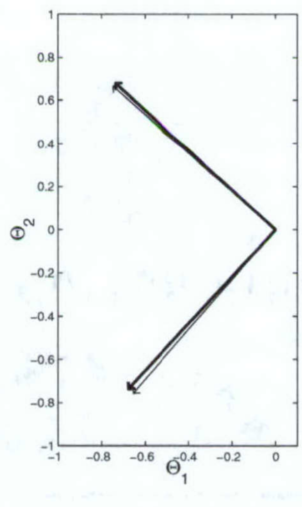

Figure 4-12: SIM Plot showing eigenmodes of original system (thick arrow), and oscillator driven system (thin arrow). The mode is plotted as a vector in Θ_1, Θ_2 space. The arrows are close together, indicating that the oscillator solution is close to the resonant mode of the underlying mechanical system.

This behavior is robust, both to changes in oscillator and system parameters as shown in figure 4-13. It is also accurate. The oscillator drives relative to the motion of the system and accurately tracks the resonant mode. If the resonant mode of the system is the desired motion (as in the crank turning case described in the following section), then the oscillator behavior is automatically synchronized with the correct motion for the task.

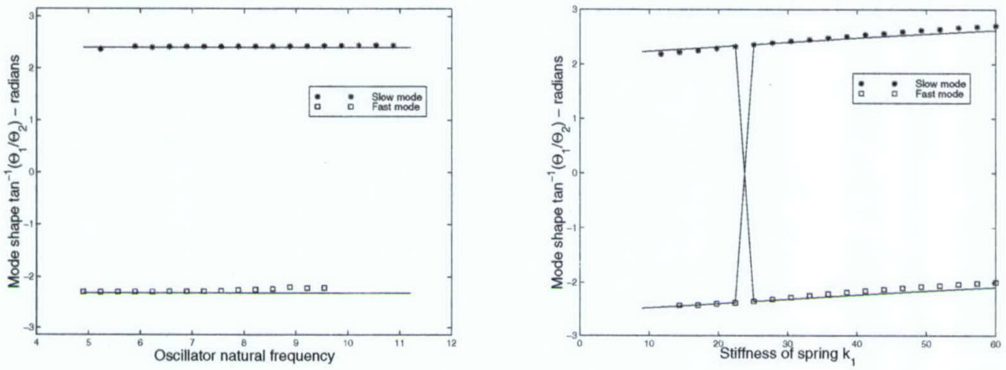

Figure 4-13: SIM Plot showing robustness of oscillator to finding mode of system, under changing oscillator and system parameters. The y axis of the plots shows the angle of the mode in Θ_1, Θ_2 space, i.e. $\arctan(\Theta_1/\Theta_2)$. The left hand plot shows the effect of varying the natural frequency of the oscillator, with the lines indicating the underlying mechanical system modes. The accuracy of the oscillator is remarkable, given that the oscillator frequency doubles over the plot range. The right hand plot shows the effect of altering the stiffness of k_1, while keeping the other stiffness constant at $k_2 = 25$ Nm/rad. When k_1 is close to 25 Nm/rad the error in the modes is very low. As the stiffness gets bigger or smaller, the approximation $k_1/m_1 \approx k_2/m_2$ becomes less accurate, and the error gets bigger. However this is a minor effect. The robustness of the oscillator system is remarkable, given the change in stiffness of six times ($k_1 = 10 \leftrightarrow 60$ Nm/rad).

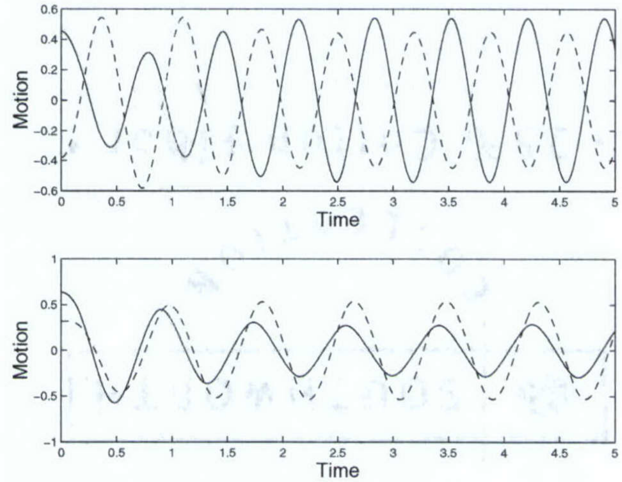

Figure 4-14: SIM Plot showing the motion of mass 1 (solid) and mass 2 (dashed) of the oscillator driven system under two different initial conditions. The oscillator converges on one of the two modes of the system, with no mixing or superposition.

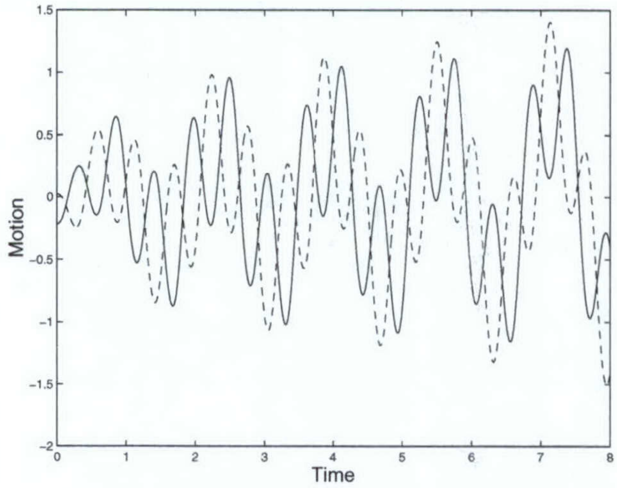

Figure 4-15: SIM Plot showing transient of the two mass system when a Van der Pol oscillator is used. This oscillator was described in section 3.8. The plot shows the motion of mass 1 (solid) and mass 2 (dashed) over time. Unlike the Matsuoka oscillator, this oscillator can give final periodic solutions which consist of a mixture of frequencies, as shown in the traces.

4.3 Local stability

The local stability of the oscillator limit cycle motion can be easily determined by examining the effect of amplitude changes on the damping in the system. At steady state, the oscillators exactly cancel the damping in the system, giving a constant amplitude oscillation. If an increase in amplitude results in positive (dissipative) damping, and a decrease in amplitude results in negative (excitatory) damping, then the limit cycle is stable.

The damping term is given by the diagonal elements of the system matrix:

$$c_{tot} = 2\zeta\omega - \omega_{ni}\mathrm{Im}[g_i e^{j\phi_i}] \tag{4.13}$$

Figure 4-16 shows $g_i e^{j\phi_i}$ plotted together with the line $j2\zeta\omega/\omega_{ni}$, for an example frequency $\omega = 7.8$. The steady state solution is marked with a box. If the amplitude of the oscillation increases, $\mathrm{Im}[g_i e^{j\phi_i}] < 2\zeta\omega/\omega_{ni}$, so $c_{tot} > 0$ which corresponds to positive damping, reducing the size of the oscillation. A decrease in amplitude has the opposite effect, confirming that the steady state oscillation is locally stable.

4.4 Summary of conclusions from the model

To conclude, the periodic solutions which exist for a set of oscillators driving a spring-mass system have the following characteristics

- The solutions are stable steady state solutions, with the oscillator canceling out the damping in the original system.
- The modes of the system are close to the eigenmodes of the original unactuated system, over a wide range of oscillator and system properties.

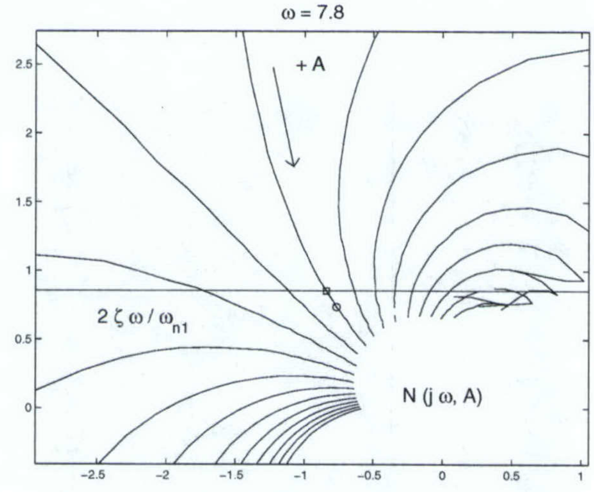

Figure 4-16: SIM Figure showing $N(j\omega, A)$ and the line $j2\zeta\omega/\omega_{ni}$ to indicate the stability of the oscillator driven system. The damping term is given by $c_{tot} = 2\zeta\omega/\omega_{ni} - \text{Im}[g_i e^{j\phi_i}]$, which is zero at the steady state solution marked by a □. An increase in the amplitude of the motion moves the system to the point marked by the ○, for which point the damping is positive because $2\zeta\omega/\omega_{ni} > \text{Im}[g_i e^{j\phi_i}]$. The positive damping will make the amplitude decrease. Similarly, a perturbation to lower amplitude results in negative damping, causing an increase in amplitude. The periodic solution is thus stable.

- The frequency of the final solution is not the eigenfrequency of the original mode, but is given by the interaction of the oscillator and system as defined by equations (4.6), (4.9).

The oscillator behavior of finding the resonant mode is useful because in some tasks such as crank turning the mode is exactly the desired motion for the task. The oscillators thus automatically find the correct coordination with the task. This complex property emerges from the interaction of the oscillators with the dynamics of the arm. No calibration, very little tuning, and no kinematic calculations are required to drive the arm along its resonant mode.

The following sections will show that the robot performance during both the crank turning and the Slinky toy tasks fit well with the simple model, providing evidence that exploiting the resonant mode is useful in practice.

4.5 Crank turning

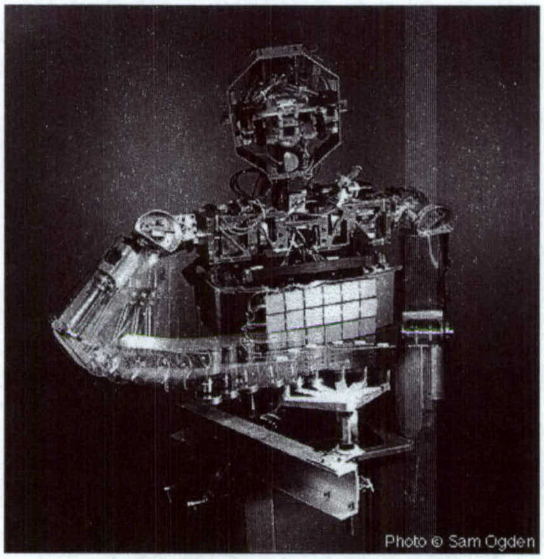

Figure 4-17: [
Picture of Cog turning a crank]Picture of Cog turning a crank. The robot is using two shoulder joints and two elbow joints to create the motion of the crank.

The crank turning behavior is a good example of the oscillators coordinated through the natural dynamics. The task is illustrated in figure 4-17. The crank turning can be understood in the context of the analysis of the previous chapter as a resonance of the springy arm around the constraint of the crank. The "springs" between the joints in this case are highly non-linear, but can be approximated by considering the solution of one mode of the system. This section compares data from the robot crank turning to the simple model developed in the previous section.

The full dynamics of crank turning are complicated and non-linear, but have the same general form as the full arm dynamics:

$$M(\Theta)\ddot{\Theta} + C(\Theta)\dot{\Theta} + K(\Theta)\Theta = K'\Theta_v \tag{4.14}$$

When this system is undergoing steady state oscillation it can be approximated a linear system, written as a set of eigenvalues and vectors. Transforming the variables $\Theta = Uq$, where U is the matrix of eigenmodes, and writing the effect of the oscillator as a diagonal matrix $G\exp(j\Phi)$, this equation can be written

$$U^T MU\ddot{q} + U^T CU\dot{q} + U^T KUq = U^T K' Ge^{j\Phi} Uq \tag{4.15}$$

Which looking at one mode u_i reduces to the single equation:

$$u_i^T Mu_i\ddot{q}_i + u_i^T Cu_i\dot{q}_i + u_i^T Ku_i q_i = u_i^T K' Ge^{j\Phi} u_i q_i \tag{4.16}$$

or approximately

$$\tilde{m}\ddot{q}_i + \tilde{c}\dot{q}_i + \tilde{k}q_i = u_i^T K' Ge^{j\Phi} u_i q_i \tag{4.17}$$

The accuracy of this as a model of the crank turning problem can then be assessed by comparing real robot data to the predictions of the model.

The crank turning performance was measured for the robot in a configuration similar to that shown in figure 4-17, using four oscillators to actuate both shoulder, and both elbow joints. The oscillator time constants and the arm stiffness were varied to examine the behavior over a range of conditions. The frequency of the motion was calculated using a zero-crossing detector, and the amplitude and phase of the various joint motions and oscillator outputs was measured using a single frequency Fourier transform. The mode of the system u_i was directly calculated from the amplitudes and phases of the joint motions, and the gain of the oscillator directly measured by comparing joint motions and oscillator outputs. The stiffness and damping at each joint was also measured.

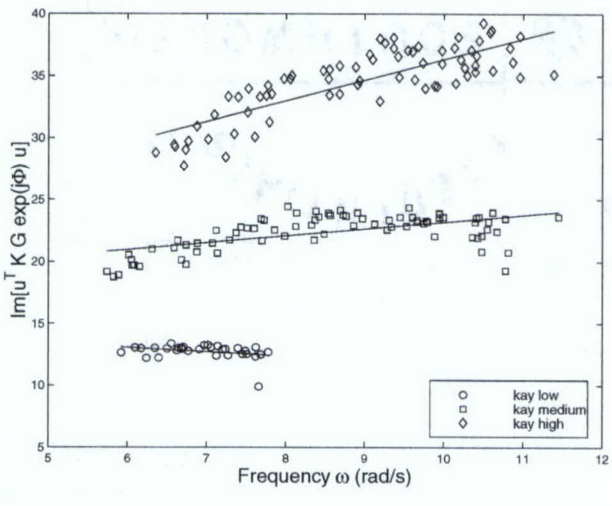

Figure 4-18: REAL Plot of $\text{Im}[u_i^T K' Ge^{j\Phi} u_i]$ against ω for the crank turning. The crank turning involves four joints of the arm, for which the mode u_i is calculated from measured amplitude and phase data. The three lines correspond to different values of the arm stiffness (○ low, □ medium and ◇ high stiffness). The theory predicts that these points should lie on straight lines, the slopes of which are detailed in table 4.1.

The model of the crank turning predicts that the effect of the oscillator is to cancel out the damping in the system, or

$$\tilde{c}\omega \approx \mathrm{Im}[u_i^T K' G e^{j\Phi} u_i] \tag{4.18}$$

Figure 4-18 shows a plot of the imaginary part of the oscillator behavior versus ω. The data points were taken by measuring the system behavior as the natural frequency of the oscillatory was altered at three different values of arm stiffness (scaling the stiffness of all the arm joints in the ratio 1 : 2 : 3). The real damping at each joint was kept constant.

The theory predicts that this should be a straight line with slope proportional to $\tilde{c}$. The lines are straight, which is itself a good result given the simplicity of the model and the complexity of the arm motion. The slopes of the lines (listed in table 4.1) increase with stiffness, indicating that there is some coupling between the stiffness and damping through the arm dynamics. The abscissa are roughly constant which is as predicted.

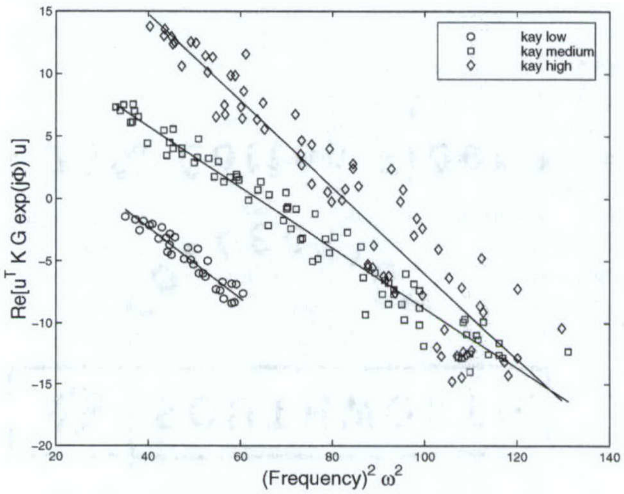

Figure 4-19: REAL Plot of $\mathrm{Re}[u_i^T K' G e^{j\Phi} u_i]$ against ω^2. The theory predicts that these lines should be straight, with approximately constant slope. The lines have gradients $-0.28, -0.24, -0.35$ for increasing stiffness which are indeed roughly constant. The line for high stiffness is the most different from constant, and also the most noisy. The abscissa are at $9.08, 15.52, 28.64$ so roughly proportional to stiffness, as predicted by the model.

The theory also predicts that the real part of the oscillator should be inversely related to ω^2:

$$\tilde{k} - \tilde{m}\omega^2 \approx \mathrm{Re}[u_i^T K' G e^{j\Phi} u_i] \tag{4.19}$$

Figure 4-19 shows the plot for this situation. Again the straight lines are striking. The mass matrix of the system $\tilde{m}$ should be independent of the stiffness. The data supports this, with slopes which are roughly constant (see table 4.1). The abscissa reflect the effect of $\tilde{k}$ in the equation above, being roughly proportional to the stiffness.

The accurate fit of the crank turning data with the model shows that the model is a good description of the overall system. It also shows that the crank turning can be thought of as a resonant mode of the arm-crank system, which the oscillators are tuning into and exciting. This perhaps explains why the crank turning generalized so easily to multiple arms, the crank turning

	$\text{Im}[u_i^T K' G e^{j\Phi} u_i]$				$\text{Re}[u_i^T K' G e^{j\Phi} u_i]$			
	Slope		Abscissa		Slope		Abscissa	
kay	Value	Norm	Value	Norm	Value	Norm	Value	Norm
1	-0.34	**1**	15.13	**1**	-0.28	**1**	9.08	**1**
2	0.56	**-1.62**	17.63	**1.17**	-0.24	**0.86**	15.52	**1.71**
3	1.66	**-4.85**	19.67	**1.30**	-0.35	**1.22**	28.64	**3.15**

Table 4.1: REAL Slopes and abscissa of the line fits in figures 4-18, 4-19. The raw values are shown in normal type, the normalized values in bold.

being the resonant mode of a larger system. It also suggests that the coupling through the Slinky toy also enforces a resonant mode of the two arm system.

The mode-finding property suggests that the oscillators would be suitable for other tasks where the resonance of the mass-spring system corresponds with the task. For example, other constrained tasks such as pumping a bicycle pump fall into this category. Methods to extend the oscillators to cases where the resonant mode of the system is not aligned with the task are described in chapter 5.

The oscillator behavior is useful because they automatically find the mode of the system without any extra computation, and they respond to the dynamics of the arm itself. The final solution is robust, because the oscillators are augmenting a natural motion of the arm.

The oscillator control is in fact more robust at high speeds, and does not work well at low speeds, which is the opposite of traditional robot control. For example, consider turning a crank using a stiff arm, with hybrid force/position control (Raibert and Craig, 1981). The arm would control force along the direction of the crank (and so exploit the physical crank to constrain the motion), and control position around the crank to move it. As the crank is turned faster and faster, the performance of both the position and force control will deteriorate as the dynamics of the arm become significant, requiring extra dynamical models. While intersegmental dynamics might change the exact structure of the crank-arm resonant mode, changing frequency will not change the fundamental structure of the dynamics. There will still be a resonant mode that the oscillators can find, and in any case the oscillators are responsive to the exact structure of the mode. By exploiting rather than canceling the arm dynamics, the oscillator system works well as the frequency increases.

4.6 Self-organization to find resonant mode

The oscillator system was always observed to find a rhythmic solution in the crank turning case. This corresponded to turning the crank all the way round over a wide set of parameters (see section 4.7), or to turning the crank part of the way round if the parameters where not set correctly. The coupling model predicts the existence of limit cycles and their local stability, but does not guarantee that the oscillators will find the solution. Unfortunately it is difficult to prove that the system will converge to the limit cycle (this is discussed further in chapter 6).

An intuitive explanation for the oscillator self-organizing behavior can be given in terms of the entrained state of the system. The oscillators start with some random phasing between one another, and by interacting with the crank settle on a solution where the phases differences between the oscillators are correct for the crank to be turned. The phases between the joints are varied by the oscillators falling in and out of entrainment with the arm.

At the steady state solution, all the joints need to move at the same frequency, with all the oscillators entrained. For the oscillator to be entrained, there needs to be a steady input, which would be given by the joint moving rhythmically. This means that there will only be a globally stable solution when all the oscillators have phases relative to one another so that they all get steady rhythmic signals from their joint angles. This situation corresponds to the crank being turned. If one of the oscillators does not get a steady input, it will become unentrained and its frequency will

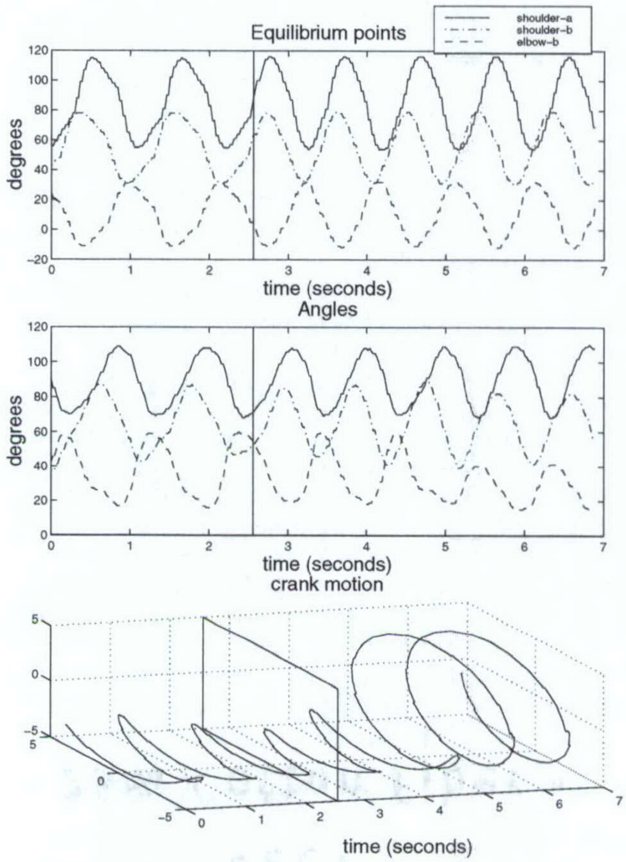

Figure 4-20: REAL Transient of oscillator driven crank turning. The three plots show equilibrium points, joint angles and a schematic of the crank motion plotted against time. The feedback to the oscillators is controlled by the step trace, and the crank angle is indicated by the dashed line. When the feedback is turned on the oscillators at the different joints entrain to different frequencies causing the phases between the joints to change rapidly. When the stable turning phase is reached, the oscillators lock onto that phase, all the oscillators oscillate at the same frequency and the crank is turned stably.

change. The different frequency has the effect of rapidly varying the phases between the joints. When the phases between the joints becomes correct, the oscillators can stay entrained, and lock onto the crank motion. The process of falling into and out of entrainment thus allows the oscillators to change the phases between the joints, so finding the solution predicted by the describing function analysis.

Figure 4-20 shows a particularly long transient for the crank turning. When the feedback is switched on, the oscillators change their speed, which causes the phases between the different joints to vary. The elbow (dashed line) speeds up and then slows down when the system converges on the limit cycle motion.

4.7 Robustness

The oscillator solutions for crank turning are remarkably robust to choices of parameters. Crank turning can be achieved for a wide variety of different parameters, and a single set of parameters is appropriate for a wide range of system dynamic properties. The oscillator parameters can be divided into those which are extremely robust (values can be varied by factors of 3 or so without affecting performance), and those which are more sensitive (values can be varied by 15-20% without affecting performance).

The most robust parameters are the arm stiffness and inertia, oscillator time constants and oscillator feedback gains. Figure 4-21 shows the range of different arm stiffnesses and time constants which were tested on the robot. Both stiffness and time constants can be changed by factors of 3 without affecting performance. The feedback gain could be varied from values of 80 to 250 about a nominal value of 168. Figure 4-22 shows some results for two arm crank turning (the configuration shown in figure 4-4). The data shows the average turning speed and was taken by varying each parameter in turn while keeping the other parameters constant. The graphs show the robustness of the crank turning to these changes. In yet another experiment, 1 Kg masses were attached to both the upper and lower parts of the arm during crank turning, as shown in figure 4-23. These masses effectively doubled the inertia of each link so changing the arm natural frequency by approximately $\sqrt{2}$. Unfortunately no data was taken, but anecdotally the only effect was to reduce the speed of the crank turning slightly and not effect the coordination of the task.

Intuitively, the reason for this robustness is the entrainment properties of the oscillator. The final frequency of the crank depends on the arm stiffness and inertia as well as the oscillator frequency. Because the oscillator can entrain over a range of frequencies, the final motion can occur over a range of time constants and arm dynamic properties. The robustness to input gain is due to the oscillator properties; the oscillator output size and phase is for the most part independent of input size (see chapter 3).

The parameters which are more sensitive are the ones more tightly related to the resonant mode shape. These are the posture of the arm (about which it oscillates) and the amplitude of the joint motions (oscillator tonic parameter). These parameters have to be roughly correct: the crank turning will not work if all the joint amplitudes are zero, and the Slinky will not work if the hands are upside down. However, due to the compliance of the arm there is a significant range for these parameters which will perform the task.

Figure 4-24 shows the range of postures which gave reasonable turning for a two degree of freedom simulated arm (configuration similar to that shown in figure 4-2). The data was obtained by exhaustively testing postures near to a tuned solution. The posture of the shoulder and elbow joints can be independently varied by about 20% without affecting the crank turning. Figure 4-25 shows data from the real robot, again from the two arm crank configuration. The graphs plot the average turning speed as a function of parameter setting, with all other parameters kept constant. The posture of the shoulder has the most effect, as might be expected given its higher stiffness and it being the most powerful joint. Figure 4-26 shows the effect of varying the amplitudes at the

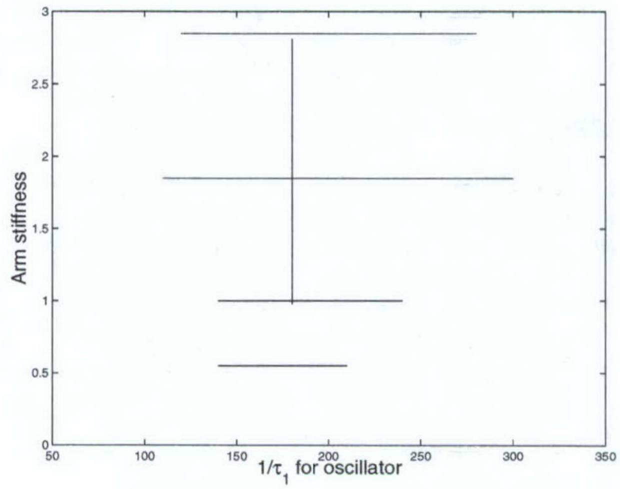

Figure 4-21: REAL Plot showing robustness of the oscillator driven crank turning to changes in arm stiffness and oscillator time constants. The lines show the extent of crank turning solutions for various experiments, and thus are samples of the underlying range of parameters possible. The ranges of working parameters was found by altering both the arm stiffness (by changing each joints stiffness by the ratio indicated from a nominal stiffness), and the oscillator natural frequency (changing τ_1 for all the oscillators). All these lines were taken with the same input scale. The range of robustness is clear, with both parameters changing by factors of 3, while still performing crank turning.

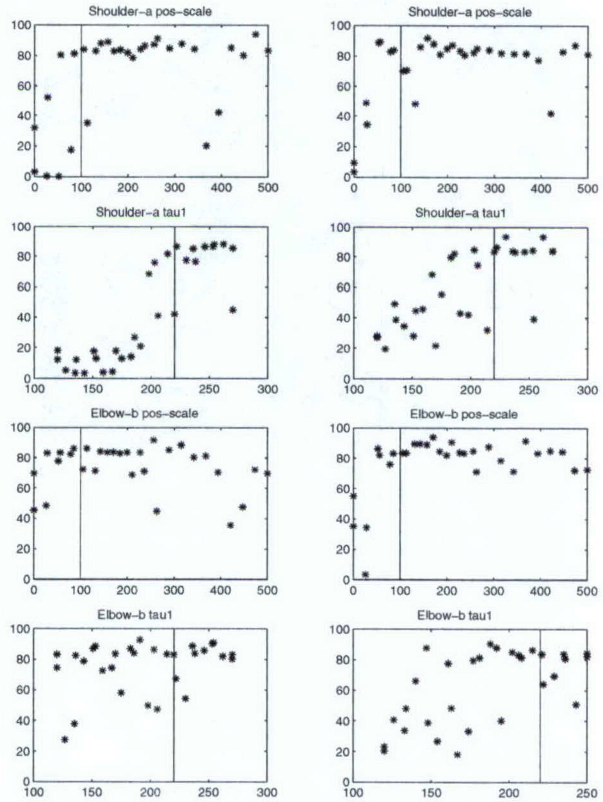

Figure 4-22: REAL The graphs show the average crank turning speed (in revs per minute) over four seconds starting from rest for two arm crank turning (the configuration shown in figure 4-4). Each graph shows the effect of varying one parameter while keeping all the others at their default values (indicated by vertical lines). The left hand column refers to changes in the left arm, and the right hand column to right arm changes. The inputs gains (marked pos-scale on figure titles) can be varied greatly without affecting performance. The time constants (marked tau1) also show some variation, requiring a minimum value for crank turning to work. The data shows the effect of varying the time constant at one oscillator while keeping all the other parameters constant, which is why the data is less robust than that in figure 4-21 where all the time constants were varied together. The ability of the oscillator to entrain over a range of frequencies is illustrated by insensitivity to time constants especially at higher values. The data was taken over a short period of time so part of the noise in the data is due to slow transients, rather than necessarily being a consequence of poor turning solutions.

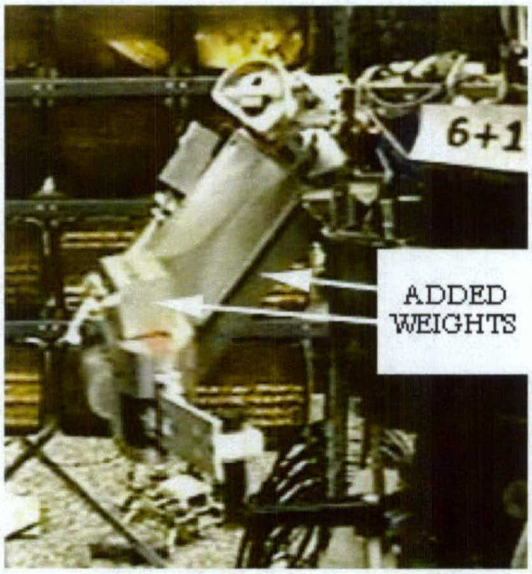

Figure 4-23: REAL To demonstrate robustness of the oscillator system, it was used with two 1 Kg masses attached to the upper and lower parts of the arm. These masses effectively double the inertia of the arm links. Their effect on the oscillator control was to make the system turn the crank slower, but did not disrupt the crank motion. In this picture all six degrees of freedom of the arm were used to turn the crank.

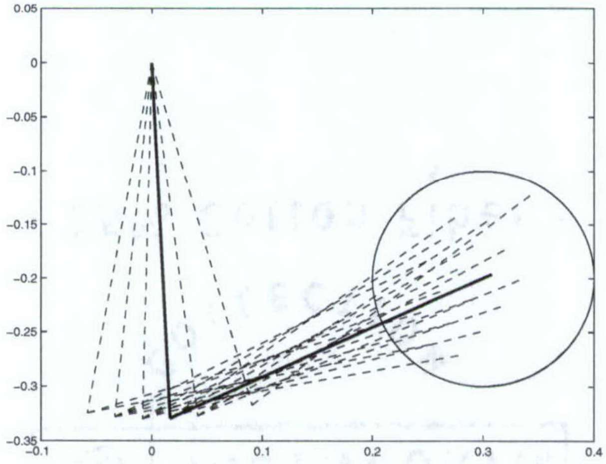

Figure 4-24: SIM The graph shows the range of postures which give good crank turning for a two degree of freedom simulated arm. The turning was measured by simulating the system and recording those postures which gave 70% of the maximum average speed over a four second transient. The plot gives a good indication of the variation in posture that can be tolerated while still giving good turning. The solid line shows the posture with the highest speed.

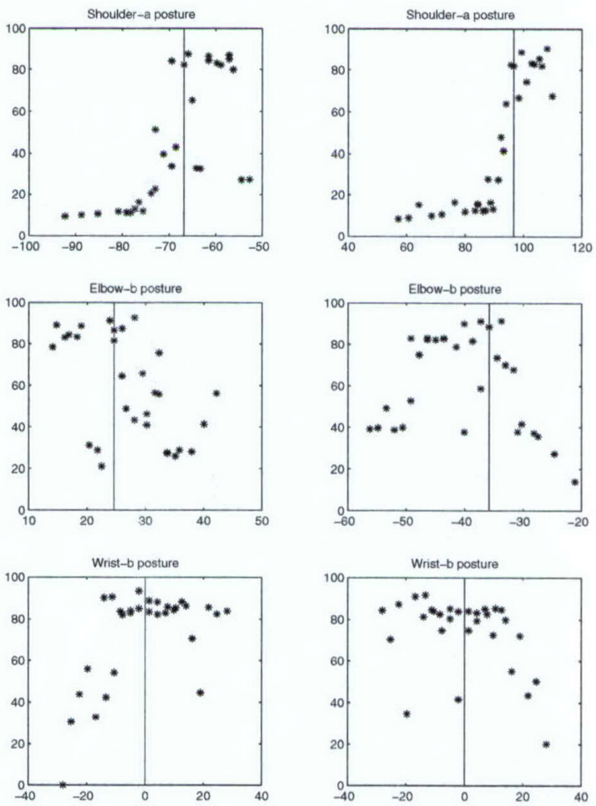

Figure 4-25: REAL The graphs show the average crank turning speed (in revs per minute) over four seconds starting from rest for two arm crank turning (the configuration shown in figure 4-4). Each graph shows the effect of varying the arm posture while keeping all the others at their default values (indicated by vertical lines). The postures are expressed in degrees. The left hand graphs are for changes to the left arm, and vice versa. The graphs indicate that the range of postures for each joint for successful crank turning is about $10° - 20°$. The data was taken over a short period of time so part of the noise in the data is due to slow transients, rather than necessarily being a consequence of poor turning solutions.

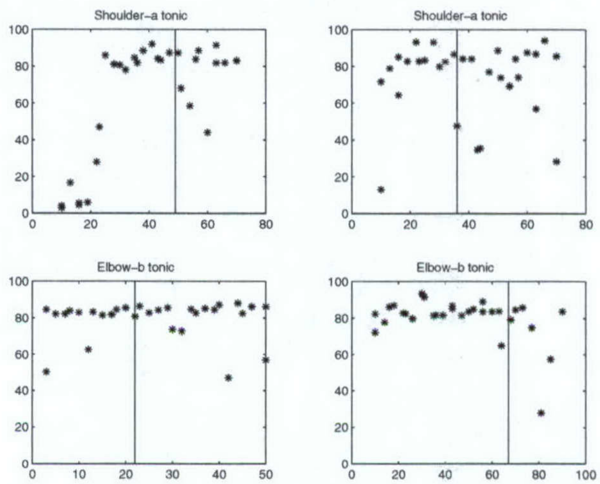

Figure 4-26: REAL The graphs show the average crank turning speed (in revs per minute) over four seconds starting from rest for two arm crank turning (the configuration shown in figure 4-4). Each graph shows the effect of varying the oscillator output amplitudes while keeping all the others at their default values (indicated by vertical lines). The left graphs refer to changes in the left arm and vice versa. The only parameter that appears to be sensitive is the shoulder-a amplitude (tonic) of the left arm. Once this parameter is large enough, the crank turning is possible for a range of parameters. The system works with large values of these parameters by turning the crank more violently, the oscillators still finding the correct coordination.

joints. The shoulder parameter is most dominant with a minimum value for crank turning. The other amplitudes do not have a strong effect. The amplitudes can be increased without affecting the crank turning because although the crank is turned more violently, the oscillators can still tune into the correct coordination to produce the turning motion.

While these parameters are more sensitive than the other oscillator parameters, they are still tolerant to significant changes. This makes tuning parameters simple. While the general form of the arm posture and the joint amplitudes are important in defining the shape of the resonant mode and thus the final driven behavior, certainly in the crank turning case their exact values are not important. This is primarily because the compliance of the arm allows errors in position to be absorbed without causing excessive internal forces or jamming.

4.8 Quality of crank turning solution

If the oscillators are finding the resonant mode of the underlying mechanical system, then that should be reflected in the energy required to produce the motion. At resonance, the real work required to move the mass should be minimized, with a considerable amount of energy stored in the springs of the arm. To investigate this, the energy used by the oscillator solution was compared to a similar solution using sine waves to actuate the joints, where the amplitudes and phases between the sine waves were chosen to be similar to the oscillator solution.

The instantaneous work done $P(t)$ by the oscillator is given by the product of the torque at the joint $\tau(t)$ and the velocity of the set-point $\dot{\theta}_v(t)$.

$$P(t) = \tau(t)\dot{\theta}_v(t) \tag{4.20}$$

or

$$P(j\omega) = \tau(j\omega)\dot{\theta}_v^*(j\omega) \tag{4.21}$$

where * is the complex conjugate. The work done $P(j\omega)$ is a complex number whose real part refers to the "active power" or work required to overcome dissipation in the system, and whose imaginary part refers to the "reactive power" or work stored and released in the springs (Bird, 1997).

The power was calculated for the oscillator and sine wave solutions, as illustrated in figure 4-27. The different graphs show the real and imaginary parts of the power for the different joints. The real parts are roughly the same, with the oscillator having a slightly higher value. For three out of the four joints, the oscillator solution has a higher reactive power. The only joint for which this is not the case is the shoulder joint, which has a small contribution to the total energy.

The ratio between energy stored and energy required is illustrated in 4-28, which shows that for three out of the four joints, the ratio is much higher for the oscillator driven system. This indicates that the oscillator driven solution stores energy in the compliance of the arm, so exploiting the dynamics of the arm to perform the task. The sine wave solution is close to the oscillator solution, but since it does not drive the resonant mode of the system, it does not produce a motion which exploits the natural dynamics of the arm as much as the oscillator solution.

4.9 Slinky toy

The second example of coordination through mechanical coupling is the task of passing a Slinky toy from hand to hand. Pictures and data for this task were included at the beginning of this chapter.

Data was collected from the operation of the Slinky in much the same way as for the crank turning. The results are included in appendix D. The main results are the same, with the data revealing the nature of the mechanical coupling through the Slinky. At low frequencies the effect of the Slinky mass provides coordination, while at high frequencies the spring-like properties of the

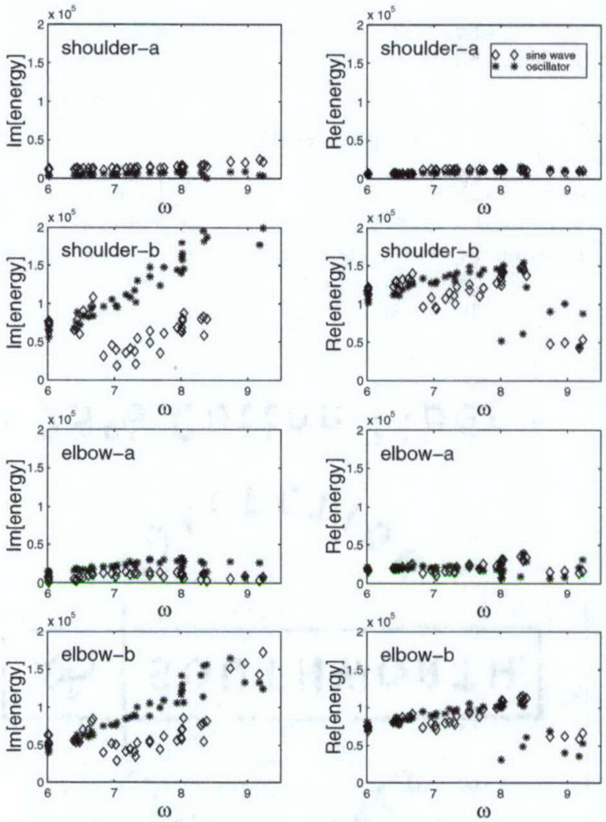

Figure 4-27: REAL Plot showing the real and imaginary parts of the power plotted against frequency for the different joints of the arm during crank turning. The imaginary part of the power (left hand graphs) refers to energy stored in the system, which is generally higher for the oscillator ($*$) than for the sine wave solution ($\diamond$). The only exception is for the shoulder-a joint, whose motion is very small, and the energy difference insignificant. The real part (right hand graphs) correspond to work done to move the arm, which is approximately the same for both control methods.

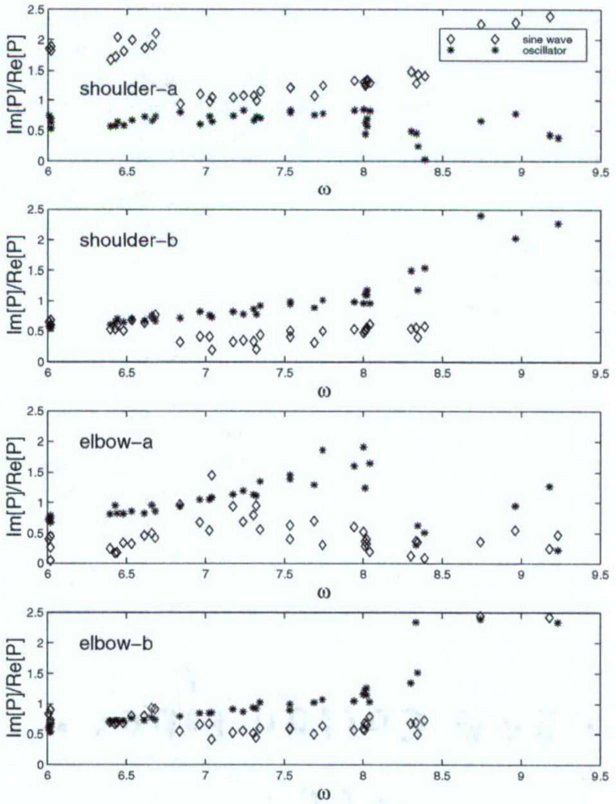

Figure 4-28: REAL Plot showing the ratio between energy stored and energy used to turn the crank, for the oscillator (∗), and the sine wave (⋄). The ratio is higher for the oscillator in all cases except for the first shoulder joint. This shows that the oscillator solution is exploiting the springy dynamics of the arm more than the sine wave solution.

Slinky have more effect. In either case the resonant mode corresponds to anti-phase motion, which is the desired motion for the Slinky.

The Slinky example is interesting because the forces on the hands due to the toy are much smaller than the forces due to the crank constraint, but the oscillators can still use the coupling to coordinate with one another.

4.10 Conclusion

This chapter has considered the behavior of the oscillators when driving multiple degrees of freedom of the arm, where the coordination between the oscillators comes from natural mechanical coupling, rather than explicit connections. The chapter presented an extension of the describing function analysis results in chapter 3 to multiple degree of freedom systems, showing that the oscillator driven limit cycles correspond to the resonant modes of the underlying mechanical system. The automatic behavior of the oscillator to find and drive systems in their resonant modes was shown to be useful in a variety of tasks, particularly for crank turning.

The oscillators are exploiting the natural dynamics of the arm in a number of ways to perform the task simply and robustly. Firstly the oscillators use the natural dynamics to couple the various joints, which removes the need for explicit connections. This makes the system more sensitive to the arm dynamics, and also reduces the dimensionality of the system, since there are less parameters to set. There is no need to specify the coordination between the links. The oscillators are also driving the arm in a natural motion, the resonant mode, which makes the overall system robust. In addition, the system exploits the constraints in the environment to constrain the movement of the arm, and so generate the movement.

The oscillator properties, together with the compliance of the arm, make the solutions robust to parameter and system changes (where parameters and stiffnesses can be multiplied by factors of 3 with no change to the system). This makes them appealing from a practical standpoint. The system quickly entrains to the task, and reacts appropriately to perturbations and changes to the system dynamics.

The oscillator solutions have also been shown to be versatile, since by actuating each joint independently the system scales easily to multiple joints, and multiple arms. Since the oscillators are simple and require virtually no tuning, scaling the system is easy. The number of tasks that can be cast as resonances is also large, especially given the sensitivity of the oscillator to both large and small coupling forces. Position constrained tasks, and ones were the manipulated object has some coupling dynamics are thus possible using this approach, as well as any other task where the resonant mode of the mechanical system is aligned with the desired motion. The method has also been extended to deal with motions which are not exactly determined by the resonant mode, these methods are considered in the following chapter.

Chapter 5

Coupling through explicit connections

5.1 Introduction

This chapter addresses the problem of extending the behavior of the oscillators to tasks which are not dictated by the natural dynamics. The problem here is constraining the oscillators in such a way that their robustness and self-organizational properties are preserved. The chapter highlights the difference between exploiting the self-organizational properties of the oscillators which are coordinated with the arm motion, and specifying constraints and relationships between the oscillators, which coordinate the oscillator outputs, not their inputs.

Three methods of constraining and modifying the oscillator behavior are considered in this chapter, illustrated schematically in figure 5-1. The first consists of adding connections between the oscillators, constraining the phase difference between the motion of the various joints. Adding connections immediately creates a tension between the oscillator entrainment with the mechanical system, and the oscillator entrainment within the network. The chapter will show that this tension is difficult to resolve, making the overall system sensitive to parameter changes and requiring tuning.

The second method uses a single oscillator to drive multiple joints. This is a generalization of the single degree of freedom systems (described in chapter 3) to multiple degrees of freedom. This method avoids some of the problems of using connections, but is still limited in its applicability. The method is illustrated by an implementation of robot sawing.

The third method relies on exploiting the resonant-mode finding behavior of the uncoupled oscillators, as described in chapter 4. The method uses extra artificially generated forces to manipulate the dynamics of the arm such that the final motion is a resonant mode of the system. The uncoupled oscillators can then find and drive the system in that mode.

5.2 Case (a): Networks of oscillators

Connecting oscillators into networks gives more complex behavior than a single oscillator because the phase relationships between the oscillators can be altered by changing the type and the strength of connections between the oscillators. Most applications of oscillators (for example in undulatory locomotion (Cohen et al., 1982), or legged locomotion (Taga et al., 1991, Kimura et al., 1998)) use oscillators connected into networks. An investigation into the stability and outputs from a variety of oscillator networks can be found in Matsuoka (1985).

Perhaps the simplest way to connect oscillators into a network is illustrated in figure 5-2, which shows two oscillators with inhibitory external connections. The connections make the input to the

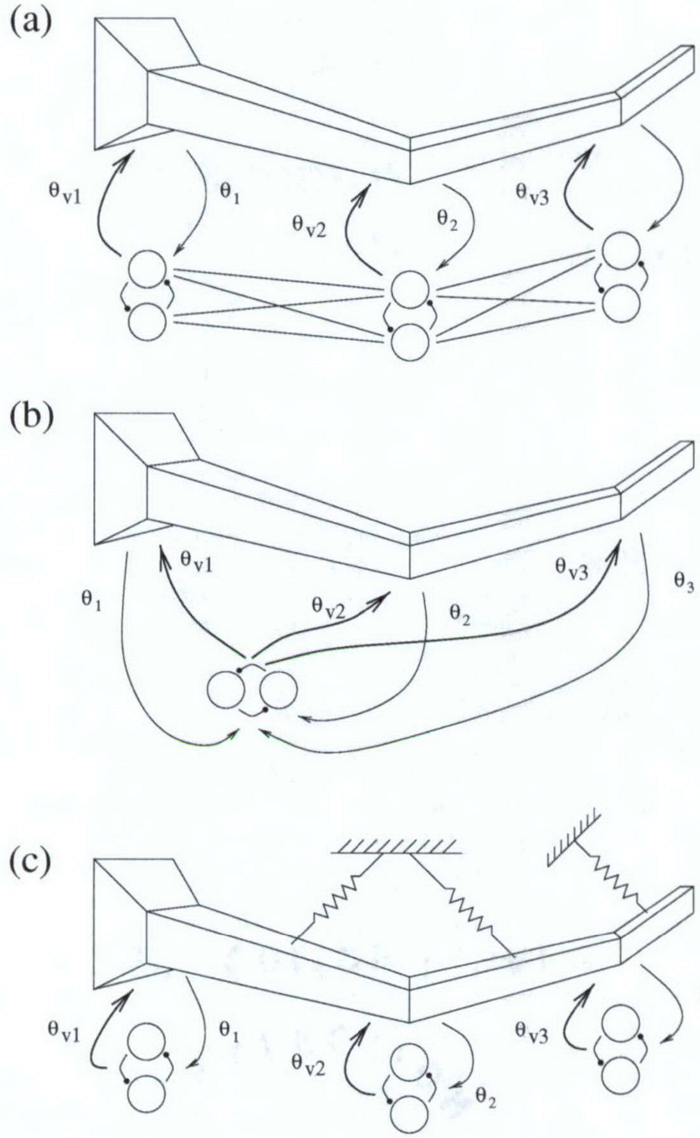

Figure 5-1: Three methods for coupling oscillators. **(a)** shows connections between the oscillators which enforce phase differences between the joints. **(b)** shows a method using a single oscillator to drive a number of joints as a single unit. **(c)** shows the case where the natural dynamics of the arm have been artificially altered by adding extra forces either from a potential field, or from virtual springs as shown here. The extra forces modify the resonant mode of the system, and so modify the final steady state solution of the oscillators. The oscillators use mechanical coupling through the augmented dynamics of the arm to tune into the resonant mode, as described in chapter 4.

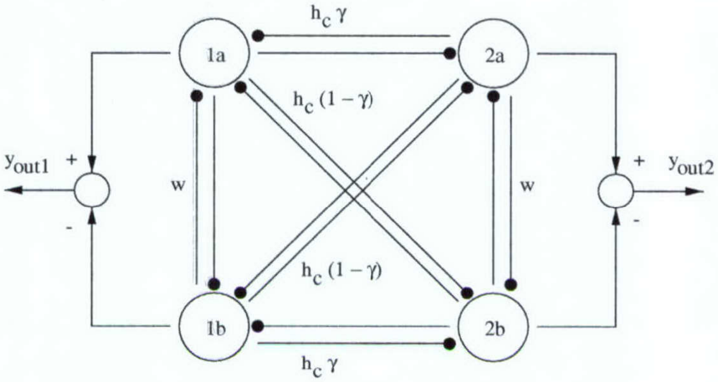

Figure 5-2: Simple oscillator network. The figure shows two oscillators with the usual mutual inhibition weights w and extra connections. The black circles on the end of the links correspond to inhibitory connections. The weights between the a neurons have strength $h_c\gamma$, and the cross coupled weights have strength $h_c(1-\gamma)$, where $0 \leq \gamma \leq 1$. If $\gamma = 1$, neuron $2a$ will fire out of phase with neuron $1a$ and so in phase with neuron $1b$. The outputs of the neurons be thus be out of phase, i.e. $y_{out1} = -y_{out2}$. Setting $\gamma = 0$ will have the opposite effect, making the outputs oscillate in phase. Setting γ to some intermediate value will result in some intermediate phase between the outputs.

neurons for oscillator 1 :

$$\text{Input}_{1a} = \Sigma_j h_j [g_j]^+ + h_c\gamma[x_{2a}]^+ + h_c(1-\gamma)[x_{2b}]^+ \quad (5.1)$$

$$\text{Input}_{1b} = \Sigma_j h_j [g_j]^- + h_c\gamma[x_{2b}]^+ + h_c(1-\gamma)[x_{2a}]^+ \quad (5.2)$$

where the first term is the usual input from external systems, $[p]^+ = \max(p, 0)$, and the parameters h_c and γ ($0 \leq \gamma \leq 1$) control the effect of the oscillators on one another. The inputs for oscillator 2 are similar. Intuitively, these connections put the neurons in mutual inhibition, causing them to oscillate out of phase. For example, if $\gamma = 1$, the connections force $1a$ to be in phase with $2b$ and $1b$ in phase with $2a$, making the outputs out of phase (see figures 5-2, 5-3).

For values of γ different from 0 or 1, the intuition is not so clear. If $\gamma = 0.5$, the input to each neuron is the sum of the outputs from each neuron of the other oscillator. Since these are out of phase due to the mutual inhibition connections internal to that oscillator (the connections marked w in figure 5-2), the actual input will be rather complicated. This results in a final phase which is dependent on initial conditions, as illustrated in figure 5-3.

Although it is straightforward to achieve phase differences of $\pm\pi$, other phase differences are difficult to achieve. The values of the gains h_c and γ need to be tuned to provide the correct behavior, and even then it is not robust to initial conditions. The behavior of this network when connected to two different mechanical systems is considered in the next section.

5.2.1 Connections and feedback

When the network shown in figure 5-2 is connected to two different mass spring systems as shown in figure 5-4, the behavior of the overall system is complicated. There is a tension between the entrainment of the oscillators with the physical systems, and the entrainment within the network. This tension makes the system sensitive to parameter changes such as the strength of the coupling and the physical properties of the mass-spring systems, as well as sensitive to the initial state of the system. The overall system is thus difficult to tune.

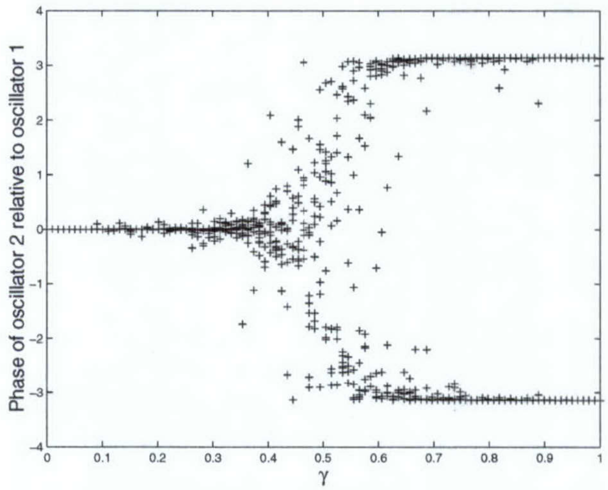

Figure 5-3: SIM Plot showing how the relative phase between the oscillator outputs varies as a function of γ. The plot was produced by simulating the system from different initial conditions for each value of γ and calculating the phase using a Fourier transform. The oscillator outputs are in phase for small γ, and out of phase for high γ. For γ's in the range $0.3 < \gamma < 0.7$ there is significant sensitivity to initial conditions, with the oscillator producing a variety of different phases.

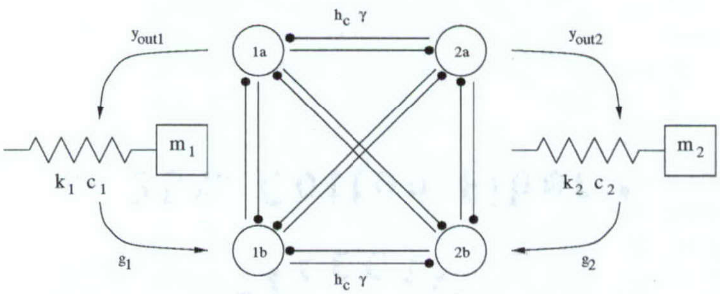

Figure 5-4: Figure showing two oscillators with external connections with parameters h_c, γ, driving two different mass-spring systems. Each oscillator is tightly coupled to a mass-spring system, with output y_{outi}, and input g_i. If the oscillators were not connected together, they would entrain with the mass-spring dynamics to reach a steady-state frequency, which would not be the same for each oscillator. If the oscillators were not connected to the mass-spring systems, then they form a coupled oscillator system which would entrain to a third different frequency. When they are all coupled together the system is forced to have one frequency, which means that some part of the system is not fully entrained.

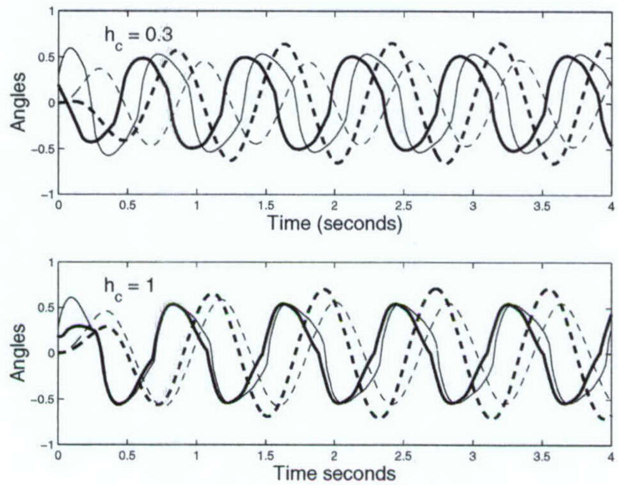

Figure 5-5: SIM Coupled system transients. The plots show the motion of the the set-points for the two masses (solid lines) and the actual mass motion (dashed), for two values of input gain h_c and $\gamma = 0$. The thick lines refer to one mass-spring system, and the thin lines to the other. This value of γ should result in the oscillators producing outputs in phase, but this is only realized for high values of the coupling strength h_c, as shown in the lower graph. The two solutions also have different frequencies, being slower when $h_c = 1$. The plot thus indicates the sensitivity of the system behavior to the values of the parameters. In addition, since the two mass-spring systems have different properties, even when the set-points are approximately in phase (lower graph), the mass motions have a phase difference between them. This means that some tuning would be required to make them move in phase.

Each part of the circuit shown in figure 5-4 would oscillate at a different frequency if isolated from the rest of the circuit. The oscillators driving the mass-spring systems would entrain to different frequencies (as described in chapter 3) and similarly the two oscillators entrain with one another through the connections at a third frequency. When the system is coupled together, each part is forced to oscillate at the same frequency, which implies that some part of the system is not fully entrained. Either the connections between the oscillators are dominant, the phase between the oscillators being set by the connection strengths and the effect of the masses largely ignored, or the mass-spring systems dominate. The strengths of the various gains and connections determine which signal dominates, which makes the overall system sensitive to parameter changes. This is in contrast to the oscillator systems analyzed in chapters 3 and 4, where changes in parameters might alter the speed of the motion, but do not radically alter the shape of it.

Figure 5-5 shows this effect. The figure shows two different responses from the coupled system for two different values of the connection strength h_c. When h_c is low, the mass motion dominates the system, and there is an appreciable phase difference between the oscillator outputs. On the other hand if h_c is large, the oscillator outputs are forced to be in phase.

The system behavior depends on the values of the gains in an inherently non-linear manner. This can be seen in the graph for $h_c = 1$ in figure 5-5. Although the oscillator outputs are in phase, the actual signals are distorted and different from one another. The distortion arises from the effect of the distorted input signals (shown in figure 5-6) on the oscillator dynamics. This behavior is more complicated than the oscillator response to a sinusoidal input (used for analysis in the previous

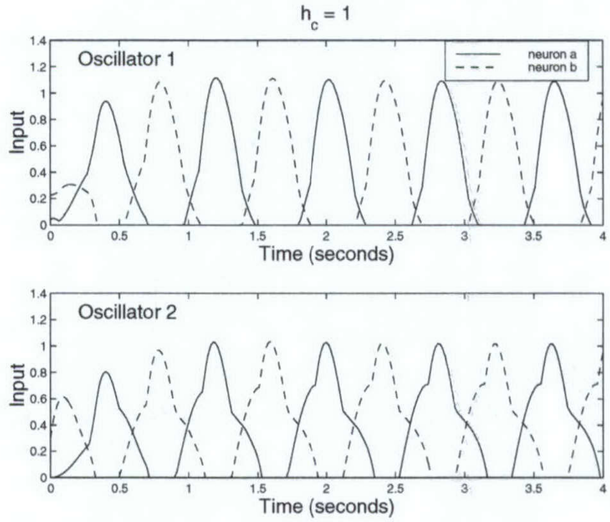

Figure 5-6: SIM Oscillator inputs. The graphs show the inputs to the individual neurons during the motion in the lower graph of figure 5-5. The top graph refers to oscillator 1, lower to oscillator 2. The inputs are not sine waves, particularly for oscillator 2 where the combination of inputs gives a very distorted signal. The distortion is created by adding the mass motion signal and the output from the other oscillator, and is one of the reasons for the sensitivity of oscillator systems coupled in this way.

chapters), and is correspondingly more difficult to predict. The input signal is itself distorted because it is a sum of approximately sinusoidal signals with different amplitudes and phases.

Even if the system could be tuned to produce outputs of a desired phase difference, that would still not be a good solution. As shown in the lower graph of figure 5-5, if the two driven systems have different dynamics, even driving them in phase will not result in in-phase motion. This is because the connections coordinate the oscillator outputs, not the oscillator inputs. Careful parameter tuning would be required to obtain the correct coordination between the masses. This contrasts with the systems analyzed in chapters 3 and 4. In those systems, the natural dynamics was used to coordinate the oscillator inputs, with the oscillator outputs driving relative to those inputs. This not only immediately gave the correct coordination with the task, but also had greater robustness to parameter and system changes than using explicit connections.

Some of these difficulties might be alleviated by ensuring that both systems have approximately equal resonant properties, and only requiring phase differences which are multiples of π. This is the case if oscillators are used for legged locomotion, since each leg has approximately equal properties, and for successful walking they must move out of phase. Nearly all of the successful implementations of oscillators with connections have been for legged locomotion (see chapter 2). For arm control, where different limbs segments have different resonant properties, and where few tasks can be easily specified in terms of phase differences between joints, connections in this way are not so appropriate. Choosing a different way of combining the inputs might also help, but since the oscillator behavior is so complicated, there is not an obvious combination method which does not introduce some undesirable non-linear effects into the system.

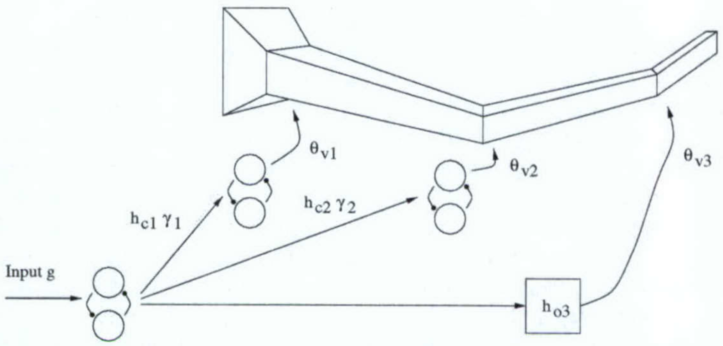

Figure 5-7: Master-slave network. A single oscillator is used to drive multiple joints of the arm using slave oscillators at the joints with explicit but unidirectional connections. Input is applied to the master, which specifies the sizes and phases for all the joints. For unidirectional connections, the effect of γ is also unreliable, with a similar behavior to that shown in figure 5-3. To overcome this, the slave oscillators can be conveniently replaced by constant gains as shown for the third joint in this picture. The magnitude of h_{o3} sets the amplitude for the third joint, with the phase (in-phase or anti-phase) set by the sign of h_{o3}.

5.3 Case (b): Master-slave network

One alternative which uses oscillators to control multiple joints of the arm is shown in figure 5-7. This uses a master oscillator to drive slave oscillators at the joints, where the connections from the master to the slaves are unidirectional. This configuration effectively makes a number of degrees of freedom operate together, under the control of one oscillator. Input to the master oscillator thus effects the motion of all the joints as a unit.

Given the unreliability of the connection strength γ (see figure 5-3), the slave oscillators can be replaced by simple gains. The magnitude of the gain determines the size of the joint motion, and the sign gives the phasing with respect to the master oscillator. The set-point for the ith joint thus becomes:

$$\theta_{vi} = h_{oi} y_o + \theta_{pi} \tag{5.3}$$

where h_{oi} is a gain multiplying the output of the reference oscillator y_o. If this gain is positive θ_{vi} oscillates in phase with y_o, and vice versa. θ_{pi} is the posture about which the oscillation occurs.

The input to the master oscillator should reflect the overall motion of the arm, as activated by the oscillator. Since the oscillator drives a number of joints as a unit, the input can be taken as a weighted average of all the joint motions. A sensible weighting factor is the size of the command to each joint, which is the gain h_{oi}. This makes the input to the master oscillator:

$$g = h_{in} \sum_i \frac{\theta_i}{h_{oi}} \tag{5.4}$$

where h_{in} is the normalization factor. Applying the input in this way gives a multiple degree of freedom generalization of the single degree of freedom system analyzed in chapter 3. By analogy, one would expect this system to have similar properties, giving a behavior similar to "resonance-tuning" as well as robustness to system properties and parameter values. In particular, one would expect this solutions to be more robust than the network considered in section 5.2.1. Although the approach is the same in that the oscillator outputs are coupled, the oscillator control ensures that these coupled outputs are synchronized relative to the input signal. They can thus respond as a whole to the system dynamics.

This configuration of oscillators was suggested by Schaal and Sternad (1998), although they suggested using oscillators at the joints rather than fixed gains, did not comment on the difficulty of producing phases, and also did not show how feedback could be applied to the master oscillator.

This configuration was implemented on the robot to perform a sawing task. A single oscillator was used to drive the two shoulder joints and two elbow joints of the arm using fixed gains. The gains were tuned to produce an approximately linear motion of the hand. A saw was attached and used to cut a 2-by-4 beam of wood. Stills from a video of this motion are shown in figure 5-8. The linear motion of the arm is not exact, however the inherent compliance of the arm allows the arm to deflect without causing the saw to jam. Because the relative phasing of the arm joints is fixed at $\pm\pi$ by the signs of the gains, the feedback to the master oscillator cannot change the relative phasing between the joints as in chapter 4. It does however alter the speed of the motion, giving a behavior similar to the resonance tuning described in chapter 3. The steady state motion thus exploits the natural dynamics by storing and releasing energy in the springs of the arm.

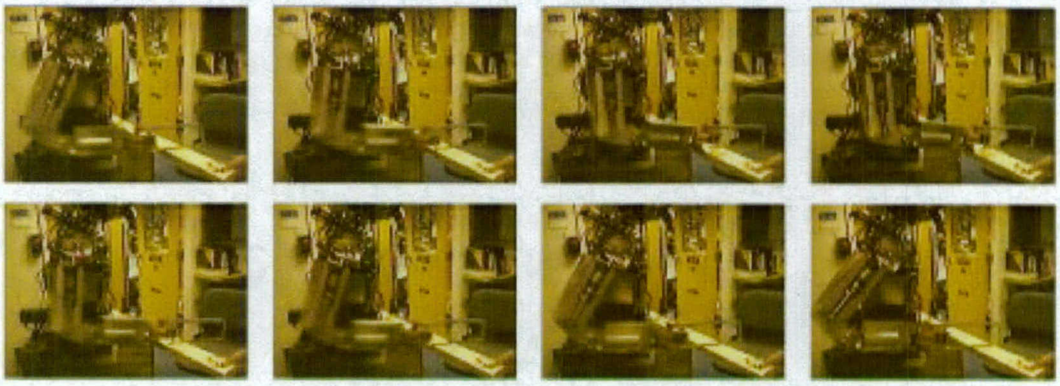

Figure 5-8: REAL Eight stills from the robot sawing a block of wood. The picture sequence runs from left to right. The pictures were taken at 0.1 second intervals and show the saw moving forward and backward, cutting the wood.

Figure 5-9 shows the energy used during sawing, plotted as a function of the oscillator natural frequency ($\propto 1/\tau_1$). The energy was calculated by integrating the instantaneous work done for all the arm joints over one oscillator cycle. The instantaneous power is defined as

$$P(t) = \tau(t)\dot{\theta}_v(t)$$

where $\tau(t)$ is the torque at the joint, and $\dot{\theta}_v(t)$ is the rate of change of the oscillator outputs. The top plot shows the real energy used in sawing, and the lower plot the "reactive" energy, the energy that is stored and released in the arm springs during the sawing. When feedback to the oscillators is used, more energy is stored in the springs of the arms over all frequencies. At low frequencies the arm does more real work per cycle, and at high frequencies the work is about the same.

The energy required for sawing does vary with frequency, due both to the damping in the arm (which is constant), and perhaps different resistances of the saw at different frequencies. The frequency under feedback is higher than without feedback, as shown in figure 5-10. There is also more variation, because the oscillator responds to the sawing resistance, cutting slower when the resistance is high, and faster when it is easier.

The energy plot is replotted in figure 5-11 as a function of the measured sawing frequency. The data is noisy, but suggests that under feedback more work is done, and more energy is stored in the joints of the arm, even for the same frequency. However, this increase in energy does not result

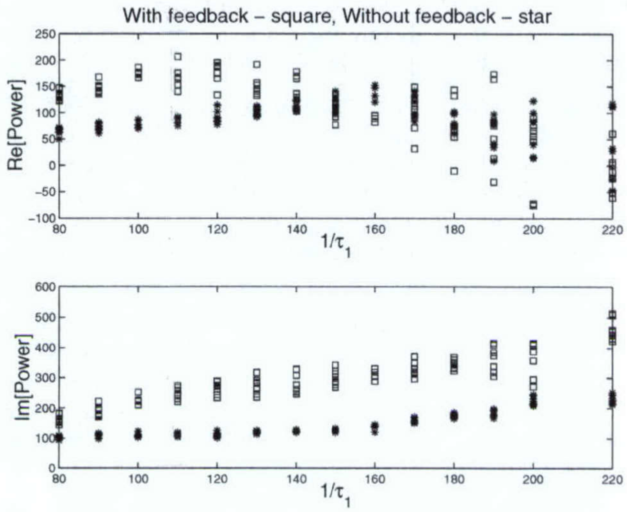

Figure 5-9: REAL Plot of energy expended in sawing versus oscillator natural frequency ($\propto 1/\tau_1$). The top graph shows the real power, i.e. the energy to overcome dissipation in the arm, and actually cut the wood, and the lower graph shows the imaginary power, which is stored and released in the springs in the arm. The □'s refer to oscillator feedback on, and the $*$'s to feedback off. When the feedback is on, the imaginary power is larger, meaning more energy stored in the arm. The real power is larger for lower frequencies, and about the same at higher frequencies. This plot is misleading because the actual frequencies of the points are different, and the energy to move the arm varies with frequency. Figure 5-11 shows the same data plotted against frequency.

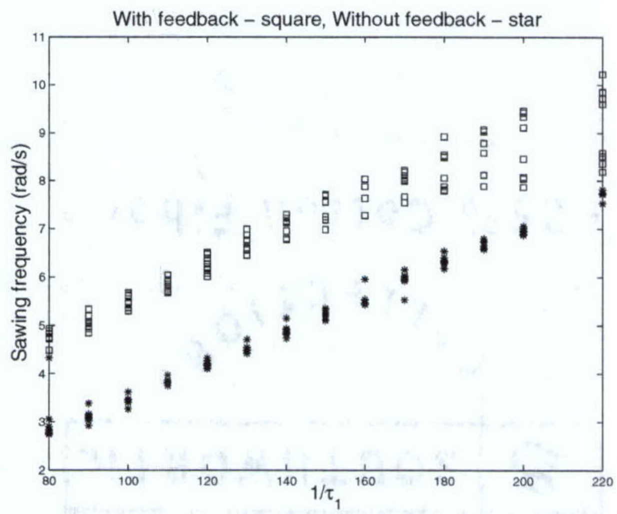

Figure 5-10: REAL Sawing frequency versus oscillator frequency for feedback on (□) and feedback off ($*$). With feedback, the sawing is faster, with a greater range of frequencies. This variation is because the oscillators respond to the varying resistance of the sawing task. When the resistance is high, the oscillators drive slower and vice versa.

in a noticeable change in the sawing efficiency, as shown in figure 5-12. This data was collected by measuring the number of strokes taken to saw though a 0.5" by 1.5" block of wood. The data is noisy, but shows that the sawing efficiency is approximately the same with and without feedback.

In summary, using feedback the arm stores more energy in the springs of the arm, requires about the same energy to move, and cuts wood at the same rate as without feedback. Since the oscillator is only scaling frequency (the coordination between the joints is fixed), one might expect the energy stored to be the same in both cases. The reasons for the discrepancy might be the slightly different amplitudes of the sawing motion under feedback, or perhaps a second order effect due to the constant phase between the drives to the joints and the weighted average of the the joint motions enforced by the oscillators.

While the effect of the feedback on the sawing performance is not particularly clear, anecdotally the feedback causes the arm to behave in a sensible manner. The arm slows when the resistance is high, and increases in speed when the resistance is low. The oscillator ensures that the command to the joints matches the joint motion, which acts to prevent jamming. In addition, if the robot is aided by a human, the oscillators can respond and adjust to a new frequency, with cooperation between the human and the robot.

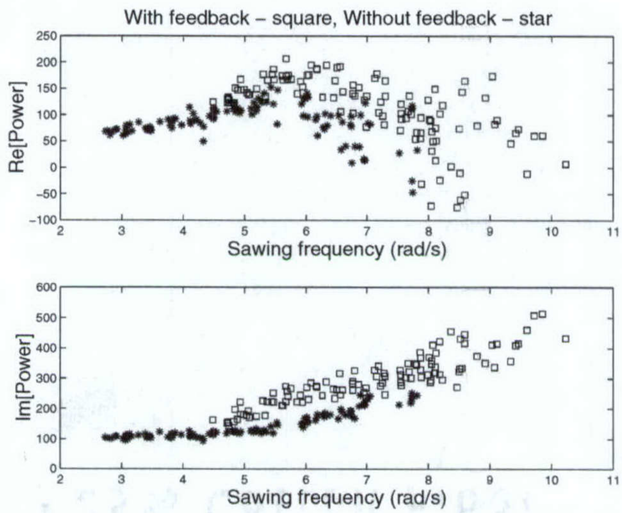

Figure 5-11: REAL Plot of energy expended in sawing versus sawing frequency. The top plot shows the real energy required to perform the sawing. The real energy is slightly higher with feedback. The lower plot shows the energy stored and released in the arm springs, which is greater for the oscillator with feedback than without it.

A simple extension of this method which would combine using connections and using mechanical coupling would be to drive different combinations of joints with separate oscillators. The coordination between these joint units or synergies could then be determined by the natural dynamics of the arm. This would allow more control of the arm motion without losing the power of coupling though the natural dynamics.

Although this method of connecting joints is more robust than the network described in section 5.2.1, it is limited in that the only possible phases are $\pm\pi$, and the overall motion is determined by rhythmic commands at the joints.