Oscillators and Crank-Turning: Exploiting Natural Dynamics with a Humanoid Robot Arm

Matthew M. Williamson
Information Infrastructure Laboratory
HP Laboratories Bristol
HPL-2002-132
June 12th , 2002*

E-mail: matthew_williamson@hp.com

robot, natural dynamics, neural oscillator, humanoid

This paper presents an approach to robot arm control that exploits the natural dynamics of the arm. This is in contrast to traditional approaches that either ignore or cancel out arm dynamics. While the traditional approaches are more general, they often result in systems and robot designs that are not robust. The alternative approach gives systems that are computationally simple, robust to variation in system parameters, robust to changes in the dynamics themselves, and versatile.

This paper examines the example of using a compliant robot arm, controlled by independent neural oscillators, in a crank-turning task. A model is constructed, and the robot behaviour compared with the model. This data shows that the arm-oscillator system is exploiting the natural dynamics by finding and exciting the resonant mode of the underlying mechanical system. Since this is a natural behaviour of the system, the robot behaviour is robust.

The paper concludes by discussing the opportunities and limitations of this approach.

* Internal Accession Date Only
Approved for External Publication

Oscillators and Crank-Turning: Exploiting Natural Dynamics with a Humanoid Robot Arm

Matthew M. Williamson
HP Labs Bristol, Filton Road, Stoke Gifford, BS34 8QZ, UK*
matthew_williamson@hp.com

Abstract

This paper presents an approach to robot arm control that exploits the natural dynamics of the arm. This is in contrast to traditional approaches that either ignore or cancel out arm dynamics. While the traditional approaches are more general, they often result in systems and robot designs that are not robust. The alternative approach gives systems that are computationally simple, robust to variation in system parameters, robust to changes in the dynamics themselves, and versatile.

This paper examines the example of using a compliant robot arm, controlled by independent neural oscillators, in a crank-turning task. A model is constructed, and the robot behaviour compared with the model. This data shows that the arm-oscillator system is exploiting the natural dynamics by finding and exciting the resonant mode of the underlying mechanical system. Since this is a natural behaviour of the system, the robot behaviour is robust.

The paper concludes by discussing the opportunities and limitations of this approach.

1 Introduction

This paper presents an approach to robot arm control that exploits the natural dynamics of the arm and its environment. The idea is to "let the physics do the computation". This results in a system that is versatile, robust, computationally simple and easy to implement. The paper describes a case study of this approach, analysing crank-turning using a compliant robot arm.

The approach is motivated by related work in robotics. A number of researchers have built systems that exploit natural dynamics to perform tasks in simple ways (e.g. 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)). It is also motivated by evidence from human movement: humans exploit the spring-like properties of their legs while running (Alexander, 1990), when swinging their arms (Hatsopoulos and Warren, 1996) and when throwing (Bingham et al., 1989)[1].

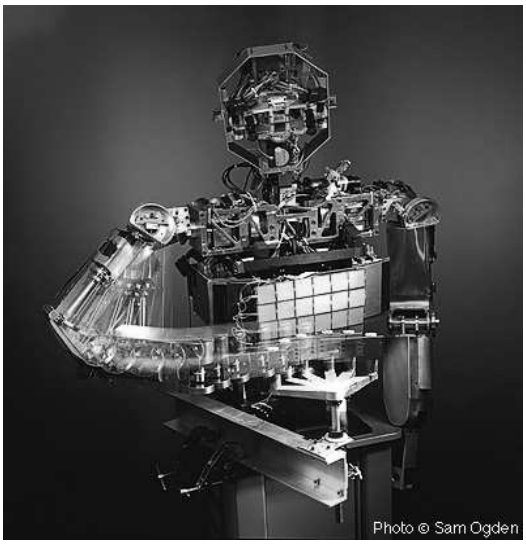

Figure 1: Picture of Cog turning a crank. The robot is using two shoulder joints and two elbow joints to create the motion of the crank.

The idea is to align the passive dynamics of the arm with the task, such that the natural behaviour of the arm is to perform the task. The motion is then achieved by using a controller to inject energy into the system.

The experiments described in this paper were conducted using the arms of the humanoid robot Cog at MIT (Brooks and Stein, 1994), shown in Figure 1. The arms are constructed using special actuators that ensure a smooth compliant motion, and are controlled by "neural oscillators" (simulations of neurons in oscillatory circuits). A great many different tasks have been demonstrated by applying the idea of exploiting natural dynamics on this system (Williamson, 1998, 1999). These include tuning into the resonant frequencies of the arm itself, juggling, turning cranks with one or two arms

*Work carried out at the MIT Artificial Intelligence Lab, 545 Technology Square, Cambridge MA 02139, USA

[1]More detailed biological evidence can be found in Williamson (1999)

(as shown in Figure 1), passing a slinky toy from arm to arm, sawing, hammering and playing drums.

For most of these tasks, the joints of the arm are controlled independently, with no connections between the joints, as illustrated in Figure 2.

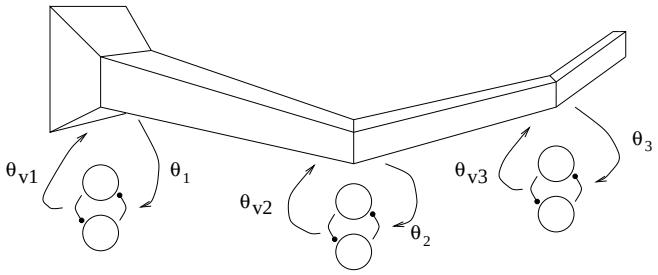

Figure 2: Figure showing configuration of oscillators and arm. The oscillators are tightly coupled to each joint, with their output driving the setpoint θ_{vi}, and their input being either the joint angle θ_i, or the joint torque u_i. There are no software connections between the oscillators; they use mechanical coupling through the physical arm to coordinate with one another and the task.

The oscillators drive the setpoint (desired position) of each joint, causing the arm to oscillate. They respond to the local dynamical state using feedback either from the joint angle or the joint torque. They thus form a tightly coupled dynamical system. Since there are no connections between the oscillators, mechanical interactions between the arm joints, detected via the feedback signals, are used to provide coordination.

The paper will compare traditional control methods with exploiting natural dynamics, showing that traditional control is more general, but exploiting natural dynamics has advantages for particular tasks. The paper will then describe in detail a crank-turning example, comparing data from the robot with a mathematical model. This will show that the robot exploits its natural dynamics to turn the crank. The paper concludes by discussing the approach of exploiting natural dynamics, highlighting opportunities and limitations.

2 Comparisons

The main difference between traditional robot control and the approach taken in this paper is the role of the robot dynamics. In traditional control, the robot is viewed as a general purpose manipulator that 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 cancelled, and certainly do not play a part in how the task is planned. The approach taken in this paper 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 behaviour 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 3. 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 behaviour 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.

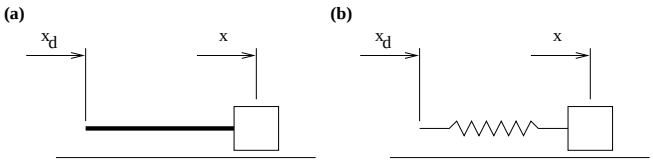

Figure 3: Comparison between traditional control **(a)**, and exploiting natural dynamics **(b)** for the task of moving a mass backwards and forwards. Under traditional control the dynamics of the robot are not used. The controller enforces a stiff structure which makes the actual mass position x track 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 (Craig, 1989).

The second difference is in terms of the design of the controller, and is illustrated schematically in Figure 4. 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 re-

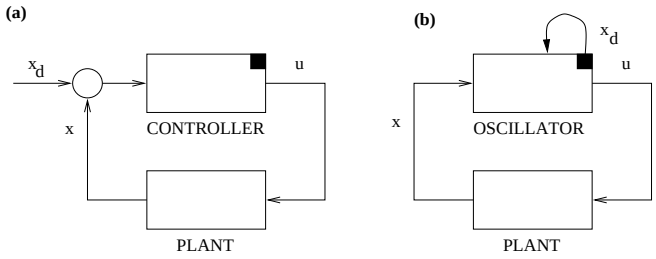

Figure 4: Comparison between traditional control **(a)**, and oscillator control **(b)**. In traditional control the plant's state x is controlled to follow the desired state x_d using a controller. This controller could have internal state as indicated by the ■, say from an integral term. The oscillator 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.

stricted 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 3, 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 synchronised 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, modelling 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 modelling 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.

3 Architecture

The arms used for this work are those of the humanoid robot Cog, as shown in Figure 1. Each arm has six degrees of freedom, and each joint is actuated by a Series Elastic Actuator (Pratt and Williamson, 1995). This actuator consists of an ordinary electric motor in series with a torsional spring. By controlling the twist of the spring, the output force of the actuator can be controlled. This arrangement gives robustness to collisions (the energy of impact is absorbed by the spring, not by the fragile gearbox teeth), is easy to make passive (important for touching objects and surfaces), and produces clean (low noise) force control.

In order to control the position of the arm, a low gain position control loop is wrapped around the force control provided by the Series Elastic Actuators. This allows the stiffness and damping at each joint to be varied, while retaining overall smooth compliant motion. The arm is moved about by altering the desired position or setpoint of the joint. The torque u_i is thus

$$u_i = k_i(\theta_{vi} - \theta_i) - b_i\dot{\theta}_i \quad (1)$$

where k_i is the stiffness of the joint, b_i the damping, θ_i the joint angle and θ_{vi} setpoint.

As mentioned above, a "neural oscillator" is used to generate the motion of the robot joints. The oscillator chosen was originally analysed by Matsuoka (1985), and consists of two simulated neurons (state variables x_i and v_i) arranged in mutual inhibition, as shown in Figure 5. The time evolution of the oscillator is given by the following equations, where $[x]^+ = \max(x, 0)$.

$$\begin{aligned}
\tau_1\dot{x}_1 &= c - x_1 - \beta v_1 - \gamma[x_2]^+ - \Sigma_j h_j[g_j]^+ & (2)\\
\tau_2\dot{v}_1 &= [x_1]^+ - v_1 & (3)\\
\tau_1\dot{x}_2 &= c - x_2 - \beta v_2 - \gamma[x_1]^+ - \Sigma_j h_j[g_j]^- & (4)\\
\tau_2\dot{v}_2 &= [x_2]^+ - v_2 & (5)\\
y_{out} &= [x_1]^+ - [x_2]^+ & (6)
\end{aligned}$$

The output of the oscillator is y_{out}, β and γ are constants[2], c is a constant that determines the amplitude of the oscillation and τ_1 and τ_2 are time constants that determine the natural frequency of the oscillator (the frequency that it will oscillate at with no input). Inputs to the oscillator are g_j which are weighted by gains h_j.

If an oscillatory input is applied to the oscillator, for a wide range of input amplitudes and frequencies the

[2]Typical values are $\beta = 2$, $\gamma = 2$.

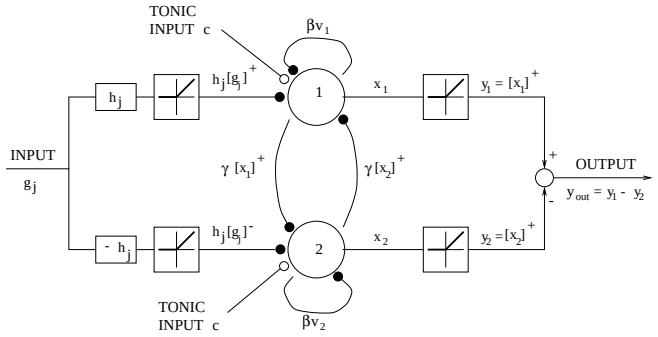

Figure 5: Schematic of the oscillator. The oscillator equations simulate two neurons in mutual inhibition as shown here. Black circles correspond to inhibitory connections, open to excitatory. The mutual inhibition is through the $\gamma[x_i]^+$ connections ($[x]^+ = \max(x, 0)$), and the βv_i connections 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.

oscillator will *entrain* the input, producing an output that is at the same frequency, but a different phase from the input. This entrainment behaviour is key to the oscillator responding the dynamics of the arm.

The oscillators are connected to the robot arm as shown in Figure 2, using the output of an oscillator y_{out} to drive the desired position of the joint θ_{vi}, and the feedback to the oscillators g_j either comes from the joint angle θ_i or the joint torque u_i.

4 Case study: Crank-Turning

The configuration for crank-turning is shown in Figure 1. In this example four joints are actuated by oscillators, the two shoulder joints, and the two elbow joints. Traces of this configuration are shown in figure 6[3].

When the feedback is not used, the oscillators produce oscillatory signals that are not coordinated with one another (they are not coupled), and so the arm moves spasmodically and does not turn the crank. However when the feedback is used, the oscillators are coupled together through mechanical coupling, and after an initial transient the crank is turned smoothly.

The crank-turning behaviour is robust to changes in most of the oscillator parameters (gains, time constants etc.) and is more sensitive to the initial posture of the arm and the sizes of the oscillator amplitudes (tonic parameter c) (see Williamson (1999) for full details). The posture and amplitudes determine whether the arm goes all way round the crank, and how violent the motion is.

[3]In order to differentiate the source of the data, data from the real robot is marked REAL and simulated data marked SIM

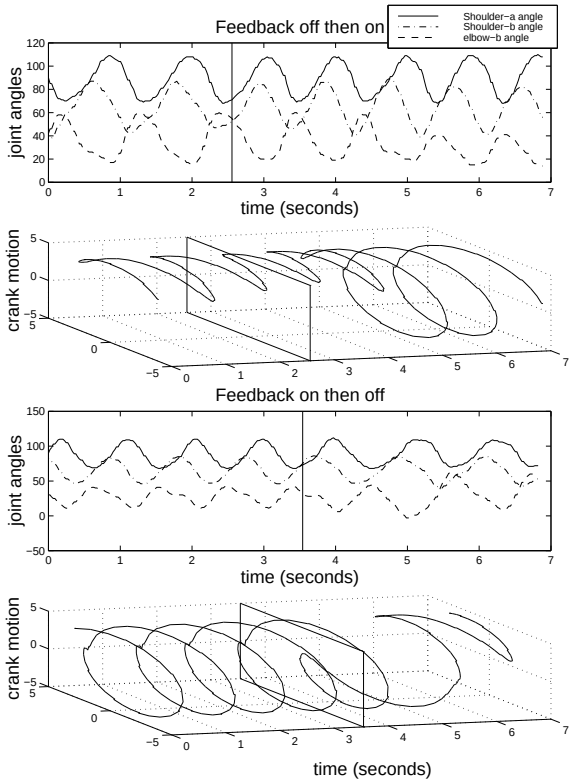

Figure 6: REAL Transients of crank turning with a redundant arm (the configuration shown in Figure 1). Three degrees of freedom are shown here, for two shoulder joints and one elbow joint. The top two graphs show the joint angles, and the motion of the crank as the feedback is turned on, which occurs 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.

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.

The cranking behaviour is also remarkably robust to different configurations of the arms, being accomplished with one arm in a redundant configuration (four and six degrees of freedom with oscillators) and in non-redundant configurations using one and two arms. It is also robust to changes in the robot system, with changing stiffnesses, damping or added weights.

Given that crank-turning is just one of an infinite number of ways that the oscillators could move the arm, answering the question "why does it work?" is important. For that we need to model the arm's oscillators and the coupling as shown in the following sections.

5 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 7. This shows two robot links represented as masses (m_1, m_2), with the joint level control appearing springs (k_1, k_2) and dampers (c_1, c_2) connected to each mass. The coupling is included as a spring (k_T) coupling the two masses, the stiffness of which can be varied to model the strength of the coupling[4]. The motion or angle of each mass is θ_1, θ_2, and the oscillator outputs are θ_{v1}, θ_{v2}.

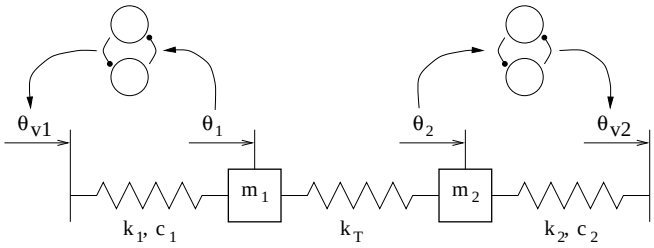

Figure 7: A simple model of coupling through the natural dynamics. The model consists of two masses driven by oscillators, connected by a coupling spring k_T.

The equations of motion for this system are

$$\begin{aligned} m_1\ddot{\theta}_1 + c_1\dot{\theta}_1 + (k_1 + k_T)\theta_1 - k_T\theta_2 &= k_1\theta_{v1} \\ m_2\ddot{\theta}_2 + c_2\dot{\theta}_2 - k_T\theta_1 + (k_2 + k_T)\theta_2 &= k_2\theta_{v2} \end{aligned} \quad (7)$$

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 resulting eigenvalue problem

$$\begin{pmatrix} \frac{k_1+k_T+j\omega c}{m_1} & \frac{-k_T}{m_1} \\ \frac{-k_T}{m_2} & \frac{k_2+k_T+j\omega c}{m_2} \end{pmatrix} \begin{pmatrix} \Theta_1 \\ \Theta_2 \end{pmatrix} = \omega^2 \begin{pmatrix} \Theta_1 \\ \Theta_2 \end{pmatrix} \quad (8)$$

for values of ω and ratios Θ_1/Θ_2.

Adding the oscillator to this analysis requires some extra work. The system is linear, but the oscillator is non-linear, and so cannot be directly added. However, when the oscillator is entrained, it produces an output at the same frequency as the input, so is effectively a linear transformation. Linear transformations are generally expressed as a transfer function $N(j\omega) = ge^{j\phi}$, where g is the gain (ratio of output amplitude to input amplitude), and ϕ the phase difference. Because the oscillator behavior depends on the input amplitude A as well as frequency, its response is a function of both i.e. $N(j\omega, A)$. This technique is known as Describing Function Analysis (Gelb and Vander Velde, 1968). For the oscillator, N can be most easily calculated by applying a variety of input amplitudes and frequencies, simulating the oscillator, and sampling the oscillator behavior.

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

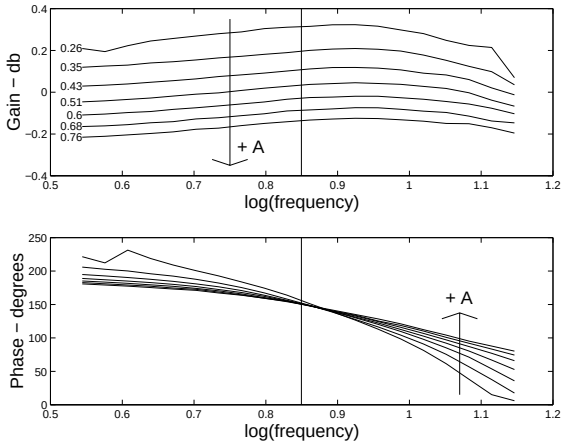

Figure 8: SIM Oscillator Bode plot. The top graph shows the gain $|N(j\omega, A)|$ plotted against frequency, and the lower plot the phase. The multiple lines correspond to different values of input amplitude A as indicated by the numbers and arrows on the plot. The gain is inversely related to A and is roughly independent of 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.

The frequency response is often plotted as a Bode plot, as shown in Figure 8. The plot shows 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 (the actual output amplitude of the oscillator is constant, making the gain $\propto 1/A$), but not the phase, which remains roughly independent of A. The gain is also roughly independent of frequency, whilst the phase is not.

Returning to the model, and writing the effect of the oscillator as $g_i e^{j\omega\phi_i}$ changes the system from being a driven resonant system (7), to a freely vibrating system:

$$\begin{pmatrix} Q_1/m_1 & -k_T/m_1 \\ -k_T/m_2 & Q_2/m_2 \end{pmatrix} \begin{pmatrix} \Theta_1 \\ \Theta_2 \end{pmatrix} = \omega^2 \begin{pmatrix} \Theta_1 \\ \Theta_2 \end{pmatrix} \quad (9)$$

Where $Q_i = k_i + k_T + j\omega c_i - k_i g_i e^{j\phi_i}$. This is solved by setting the imaginary parts to zero (the final solutions are steady state, they do not grow or decay) (Strang, 1993).

$$\omega c_i = k_i \mathrm{Im}[g_i e^{\phi_i}] \quad (10)$$

and then solving numerically for the oscillator frequency and phase. The final solution for ω can then be obtained by solving the resulting eigenvalue problem. Full details are given in Williamson (1999).

The expression for the eigenfrequencies is obtained by solving (9) and is complicated:

$$\omega^2 = \frac{\Sigma_G + k_T\Sigma_m \pm \sqrt{\Delta_G^2 + 2k_T\Delta_m\Delta_G + k_T\Sigma_m^2}}{2} \quad (11)$$

where $G_i = (k_i/m_i)(1 - g_i\cos\phi_i)$, $\Sigma_G = G_1 + G_2$, $\Delta_G = G_1 - G_2$, $\Delta_m = (1/m_1 - 1/m_2)$ and $\Sigma_m = (1/m_1 + 1/m_2)$. This can be simplified by considering that at constant frequency, the phase of N does not change greatly with input amplitude (see Figure 8), and the phase is such that $\cos\phi$ is small. This makes $\Delta_G \approx k_1/m_1 - k_2/m_2$, which can be neglected compared to the other terms in (11), giving

$$\omega^2 \approx \begin{cases} \Sigma_G/2 + k_T\Sigma_m \\ \Sigma_G/2 \end{cases} \quad (12)$$

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

$$\Theta_1/\Theta_2 \approx \begin{cases} \frac{2k_T/m_1}{k_1/m_1 - k_2/m_2 - 2k_T/m_1} \\ \frac{2k_T/m_1}{k_1/m_1 - k_2/m_2 + 2k_T/m_2} \end{cases} \quad (13)$$

which is the same as the resonant modes of the underlying mechanical system.

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 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 (10) and (11).
- The modes of the system are close to the eigenmodes of the original unactuated system, over a wide range of oscillator and system properties. This is shown in Figure 9, which shows the oscillator accurately finding the mode as either its parameters are doubled, or the system parameters are changed by a factor of six.
- The oscillator only finds one mode, not a superposition of modes as one would find in the linear case.
- The decentralized oscillator system requires no calibration, very little tuning, and no kinematic calculations to drive the system in its resonant mode.

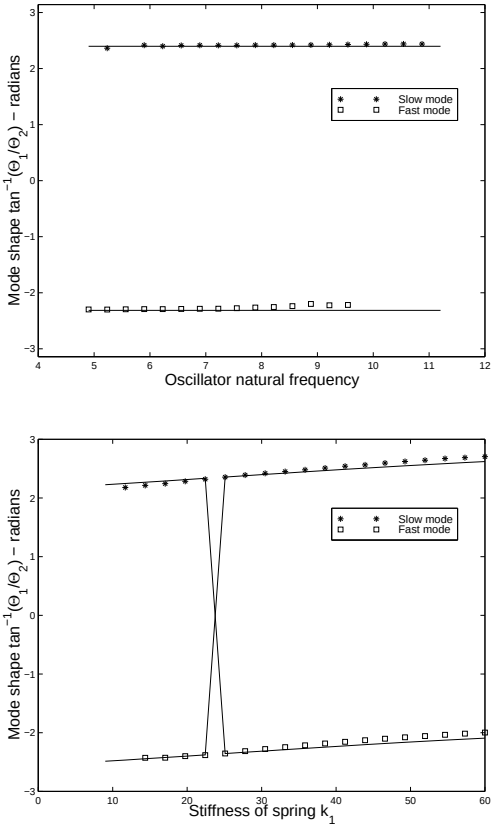

Figure 9: SIM Plot showing robustness of oscillator to finding mode of system, under changing oscillator and system parameters. The y axis of the plots shows the angle of the mode in Θ_1, Θ_2 space, i.e. $\arctan(\Theta_1/\Theta_2)$. The top plot shows the effect of varying the natural frequency of the 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 bottom 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 slightly bigger. The robustness of the oscillator system is remarkable, given the change in stiffness of six times ($k_1 = 10 \leftrightarrow 60$ Nm/rad).

6 Comparison with robot data

The previous section showed that using independent oscillators at the joints of a springy system would result in the system being driven along its resonant mode, thus exploiting the natural dynamics of the system. This is an appealing explanation for the robot crank turning: connecting the arm to the crank constrains the motion, and the lowest frequency mode (which is often the largest gross behavior) is to turn the crank. In this section we

compare the robot crank-turning to the model, to confirm that this is an accurate explanation.

The coupling springs in the robot case are rather more complicated than the linear k_T used in the analysis, but can be approximated by considering the solution of one mode of the system. The full dynamics of crank-turning are complicated and non-linear, but have the same general form as the full arm dynamics (Craig, 1989):

$$M(\Theta)\ddot{\Theta} + C(\Theta)\dot{\Theta} + K(\Theta)\Theta = K'\Theta_v \qquad (14)$$

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

$$U^T M U\ddot{q} + U^T C U\dot{q} + U^T K U q = U^T K' G e^{j\Phi} U q \quad (15)$$

Which looking at one mode u_i reduces to

$$u_i^T M u_i \ddot{q}_i + u_i^T C u_i \dot{q}_i + u_i^T K u_i q_i = u_i^T K' G e^{j\Phi} u_i q_i \quad (16)$$

or approximately

$$\tilde{m}\ddot{q}_i + \tilde{c}\dot{q}_i + \tilde{k}q_i = u_i^T K' G e^{j\Phi} u_i q_i \qquad (17)$$

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 1, using four oscillators to actuate both shoulder, and both elbow joints. 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.

To generate a range of data points, the stiffness of the arm was varied by scaling the stiffness at all the arm joints by the ratio 1:2:3 i.e. doubling and tripling the arm stiffness, keeping the damping constant. The oscillator time constants were also varied.

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

$$\tilde{c}\omega \approx \mathrm{Im}[u_i^T K' G e^{j\Phi} u_i] \qquad (18)$$

Figure 10 shows a plot of ω versus the imaginary part of the oscillator behavior. Equation (10) predicts that this should result in a straight line with gradient $\tilde{c}/\tilde{k}$. For these experiments the damping was kept constant, so $\tilde{c}$ should also be constant. The lines are straight, which is

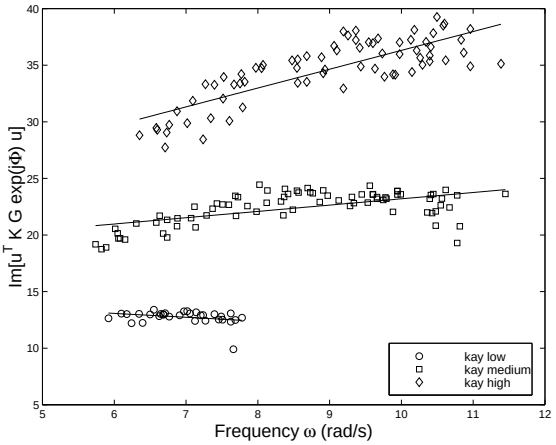

Figure 10: REAL Plot of $\mathrm{Im}[u_i^T K' G e^{j\Phi} u_i]$ against ω for the crank-turning. The crank-turning involves four joints of the arm, for which the mode is calculated from measured amplitude and phase data. The three lines correspond to different values of the arm stiffness (∘ low, □ medium and ◇ high stiffness). The theory predicts that these points should lie on straight lines with gradients proportional to the ratio of damping to stiffness in the system. The actual gradients are detailed in Table 1, and are in fact inversely proportional to arm stiffness.

kay	Value	Norm	Predict
1	-1.63	**1**	1
2	-0.85	**0.52**	0.5
3	-0.47	**0.29**	0.33

Table 1: REAL Gradients for the lines in Figure 10.

itself a good result given the simplicity of the model and the complexity of the arm motion. The gradients of the lines (listed in Table 1) are approximately proportional to $1/\tilde{k}$, which is exactly as predicted by the theory.

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

$$\tilde{k} - \tilde{m}\omega^2 \approx \mathrm{Re}[u_i^T K' G e^{j\Phi} u_i] \qquad (19)$$

Figure 11 shows the plot for this situation. Again the straight lines are striking. One might expect the mass matrix of the system $\tilde{m}$ to be independent of the stiffness, making the gradients constant. The gradients are shown in Table 2. They are roughly constant given the scatter in the plots (gradients in the ratio 1:1.14:1.8). The abscissa reflects the effect of $\tilde{k}$ in the equation above, with the abscissa increasing with increasing stiffness. The data set which gives the poorest comparison is that for kay= 3, which is the noisiest of the three cases.

The reasonably accurate fit of the crank-turning data with the model shows that the model is a good description of the overall system. It also lends support to the

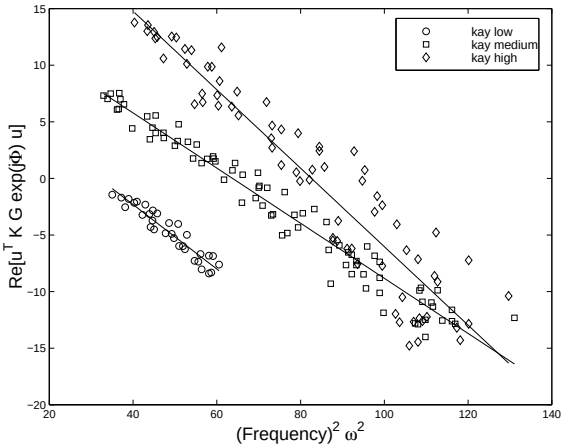

Figure 11: REAL Plot of $\mathrm{Re}[u_i^T K' G e^{j\Phi} u_i]$ against ω^2. The theory predicts that these lines should be straight, with approximately constant gradient. The lines for low and medium stiffness (kay) have roughly equal gradients, and the line for high stiffness is different, but the data is considerably more noisy.

	Gradient			Abscissa		
kay	Value	Norm	Predict	Value	Norm	Predict
1	-0.19	**1**	1	5.63	**1**	1
2	-0.21	**1.14**	1	14.02	**2.49**	2
3	-0.33	**1.8**	1	27.94	**4.96**	3

Table 2: REAL Gradients and abscissa for the lines in Figure 11.

explanation that crank-turning can be thought of as a resonant mode of the arm-crank system, which the oscillators are tuning into and exciting. The natural dynamical properties of the system are thus being exploited.

7 Discussion

This paper has presented the idea of exploiting natural dynamics in robot control. This means "let the physics do the computation", aligning the natural dynamics of the arm with the task, and using simple controllers to excite and perturb the otherwise natural motion.

Compared to traditional control techniques that use computation and robot design to eliminate robot dynamics, this approach emphasises the importance of having significant dynamics. This allows the use of compliant arms that in themselves have some advantages (stability, robustness against collisions etc.). Traditional approaches are more general, but for particular tasks, exploiting the natural dynamics gives solutions that are more robust.

These claims have been examined using the example of a compliant robot arm driven by neural oscillators for a crank-turning task. The crank-turning behaviour was modelled, demonstrating that the action of the oscillators was to find the resonant mode of the arm, a fundamental property of the arm dynamics. This was backed up by data taken from the robot. The crank-turning example shows the power of exploiting natural dynamics: complex tasks can be achieved with remarkable simplicity; the solution is robust to large changes in both the oscillator parameters and the system properties; and the resulting system is versatile.

The question remains of the relevance of this approach to other robot tasks. While some tasks (e.g. crank-turning, bicycle pumping) are well constrained, others are less (e.g. sawing), or not at all (e.g. free movement). There are three possibilities for these tasks, either to add connections between the oscillators to enforce phase relationships, (this has been shown in Williamson (1999) not to be effective), drive the arm in different ways (drive combinations of joints, not individual joints) or create an artificial potential landscape over which the arm moves. The middle approach is perhaps most biologically inspired, as our muscles tend to actuate combinations of joints, rather than single ones (Hogan, 1985).

For tasks that are not rhythmic, the oscillators can still be used, by using a single cycle of the oscillator to drive the arm, and relying on the entrainment properties of the oscillator to excite the natural dynamics. Some early work on this is reported in Williamson (1999).

More generally, the approach of exploiting the underlying behaviour of the system rather than enforcing that behaviour using computation feels intuitively powerful. While it is difficult to design systems that work in this way, the work in this paper gives some pointers. Important aspects are the physical constraints of the task (e.g. the crank), how the control is organised (at the joint level, or combinations of joints?), what aspect of the dynamics is exploited (crank-turning exploits mass-spring resonance, but the effect of gravity, or skeletal loads could also be exploited), and what controller is used (the oscillators are efficient at finding resonant modes, while other controllers might be appropriate for different situations).

This paper has shown some of the benefits of exploiting natural dynamics. Hopefully it will challenge and inspire others to try this approach.

References

Alexander, R. M. (1990). Three uses for springs in legged locomotion. *International Journal of Robotics Research*, 9(2):53–61.

Bingham, G. P., Schmidt, R. C., and Rosenblum, L. D. (1989). Hefting for a maximum distance throw: A smart perceptual mechanism. *Journal of Experimental Psychology: Human Perception and Performance*, 15(3):507–528.

Brooks, R. A. and Stein, L. A. (1994). Building brains for bodies. *Autonomous Robots*, 1(1):7–25.

Craig, J. J. (1989). *Introduction to Robotics: Mechanics and Control*. Addison-Wesley, Reading, Massachusetts, second edition.

Gelb, A. and Vander Velde, W. E. (1968). *Multiple-input Describing Function and Nonlinear System Design*. McGraw-Hill.

Hatsopoulos, N. G. and Warren, W. H. (1996). Resonance tuning in rhythmic arm movements. *Journal of Motor Behavior*, 28(1):3–14.

Hogan, N. (1985). The mechanics of multi-joint posture and movement control. *Biological Cybernetics*, 52:315–331.

Matsuoka, K. (1985). Sustained oscillations generated by mutually inhibiting neurons with adaption. *Biological Cybernetics*, 52:367–376.

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

Pratt, G. A. and Williamson, M. M. (1995). Series elastic actuators. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS-95)*, volume 1, pages 399–406, Pittsburg, PA.

Raibert, M. H. (1986). *Legged Robots That Balance*. MIT Press, Cambridge, Massachusetts.

Schaal, S. and Atkeson, C. G. (1993). Open loop stable control strategies for robot juggling. In *Proceedings 1993 IEEE International Conference on Robotics and Automation*, volume 3, pages 913–918.

Strang, G. (1993). *Introduction to Linear Algebra*. Wellesley-Cambridge Press, Wellesley, MA.

Williamson, M. M. (1998). Neural control of rhythmic arm movements. *Neural Networks*, 11:1379–1394.

Williamson, M. M. (1999). *Robot Arm Control Exploiting Natural Dynamics*. PhD thesis, Massachusetts Institute of Technology, Cambridge, MA.