Neural network control of a cricket robot

Richard Reeve and Barbara Webb
Centre for Cognitive and Computational Neuroscience
Department of Psychology
University of Stirling, Stirling.
rer2@stir.ac.uk, bhw1@stir.ac.uk

5.4.2001

Abstract

This paper advances previous work using a robot to model cricket behaviour by implementing a more sophisticated neural model. The main characteristics of the model are the use of "integrate and fire" neurons, with both voltage dependent and current injection synapses that have their own dynamics. It is designed to be fully consistent with those developed in theoretical neuroscience, while also efficient enough to control a robot in real time. Here we show how this models allows a small network of neurons to be selective to a temporal pattern in a sound source, and demonstrate its use to control a robot to track that sound source. Our aim is to use this model as a basis for building biologically-inspired systems for integrating different sensorimotor behaviours on the robot, to achieve effective sound localisation in an outdoor environment.

1 Introduction

Much previous work has been done to model the behaviour of the cricket *Gryllus bimaculatus*, using a Khepera robot to determine whether specific neural connections were sufficient to explain the insect's phonotaxis behaviour [*e.g.* Webb, 1996]. Recent work by Webb and Scutt [2000] has shown that a model using only four neurons is sufficient to explain many features of this behaviour, such as temporal pattern preference and choice between different sounds. Here, we extend that work by providing a more sophisticated model of neural processing. As well as permitting further investigation of the phonotaxis circuit, we plan to use the model for a robot that will integrate multiple sensorimotor behaviours, to perform phonotaxis outdoors on uneven terrain.

This will be implemented first on a Koala robot and then on a six-legged robot being developed at CWRU. We want to address the problem of phonotaxis under conditions that represent more naturalistically the problems faced by real crickets. This includes factors such as the large change in amplitude of the signal as the robot approaches the source, the kinds of distortion of the signal caused by the natural environment, and the problems of moving over a ground surface that makes straight-line progress difficult. We have already demonstrated the addition of an optomotor behaviour to aid maintenance of a constant heading during phonotaxis [Webb and Harrison, 2000]. We are interested in understanding how animals actually carry out the necessary integration of different sensorimotor responses, and adaptation of behaviour to context, at the level of neural circuitry. Hence we need to ensure the neural model we adopt is appropriate for addressing these issues.

We also wished to have a more principled implementation of the level of functionality that had evolved in various reimplementations of previous models. A characteristic of these models was that they represented a number of "low-level" dynamic characteristics not typically found in Artificial Neural Nets. As well as "spiking", critical properties include synaptic depression and facilitation (described further below). Our belief is that these dynamic characteristics of neurons are particularly important for understanding how multimodal and context-dependent behaviour is controlled in animals, and may suggest mechanisms for achieving comparable performance flexibility in robots. It also allows us to copy

Proc. TIMR 01 - Towards Intelligent Mobile Robots, Manchester 2001.
Technical Report Series, Department of Computer Science, Manchester University, ISSN 1361 - 6161.
Report number UMCS-01-4-1. http://www.cs.man.ac.uk/cstechrep/titles01.html

neuroethological systems at the level they are investigated in biology. At the same time we wanted to improve consistency of the model with some of the family of more formal models that have been developed in theoretical neuroscience, in particular those discussed by Koch [1999], with the hope that some of the analyses would thus carry over. The model would also then be directly compatible with analog VLSI circuits currently being developed [*e.g.* by Indiveri, 2000]. Finally we wanted to fulfill the constraints of being able to use the neural model to control robot motor responses to real sensor inputs in real time: to explore networks that are both large and relatively realistic, on modest computing resources.

Both the Khepera and Koala robots have relatively slow processors on board. Previous experiments have relied on intelligent sensors taking a lot of potential load off the robot; the incoming data are pre-processed to make them match the cricket's own sensors, and this output can than be fed directly into neurons without any further processing by the robot. Also neurons used in previous experiments were simplified (particular by the use of linear rather than logarithmic decay functions) to allow them to work on the limited processor available on the Khepera [Webb and Scutt, 2000]. However, there is still a limitation on the number of neurons that can be simulated by the processor without the speed dropping far enough to affect performance, for instance previous versions of optimised C code have been able to simulate only a handful of neurons. Chapman [2001] has taken the approach of writing the simulator in carefully optimised machine code; this was effective in allowing larger circuits (up to 100 neurons) but somewhat inflexible and hard to debug. It would be nice to use a dedicated neural chip (*e.g.* the aVLSI chips described above) but this requires a considerable lead time to design and build the chip, and is again rather inflexible — we have taken into account in the following design the possibility of substituting in a neural chip for the simulator at a later stage. But for the present we are instead adopting the approach of using a faster offboard processor. This has the additional advantage of allowing us to do "off-line" simulation of the network response to pre-recorded sensory data, which is useful for designing and parameterising the neural circuits. The system is currently implemented on a Linux workstation and can simulate more than a thousand neurons in real time updating at 1 kHz with ease.

2 Neural model

2.1 Motivation

Traditionally, neurons in Artificial Intelligence have been idealised for the sake of mathematical tractability to produce threshold and sigmoidal neurons, where the output of a neuron is a simple function of its inputs. The analytical tractability has made it particularly easy to train feedforward networks using gradient descent algorithms like backpropagation [Rumelhart and McClelland, 1986] or more sophisticated ones such as conjugate gradient descent [Bishop, 1995, ch.7], and indeed they have been used in the past for control of insect-like walking robots.

However, many low-level features of real neurons are ignored by such models. Including such features allows a richer set of behaviours to be expressed using only a small network. These include:

Internal state Neurons often need to respond to a pattern of inputs over time (*e.g.* the chirping pattern of male crickets), and this is most easily achieved by having state in the neurons. It could also be done by using recurrent sigmoidal neurons, but they are no easier to train; moreover stateless neurons does not exist in biology.

Spiking Many (though not all) real neurons spike, including those involved in auditory processing in the cricket. Neuroethologists are increasingly recognising that the spiking "code" contains much more information than that given by just the spike rate.

Dynamic and adaptive synapses Synapses in sigmoidal neurons are represented simply by a weight, which scales the output of a neuron allowing it to have different effects on different neurons. Adding state to the synapses allows the spiking inputs to be spread over time, and adaptation enables the synapses to respond differently to spikes depending on the recent history of the synapse. Again this is biologically realistic and is important, for example, in allowing the networks to respond appropriately to the specific syllable and chirp lengths in the cricket song.

Spike propagation times Spikes do not travel instantaneously from neuron to neuron in biological systems, and this feature can be exploited to allow more complex timed behaviours with fewer neurons.

These features had been added piecewise to our previous model. Here we attempt to make a more principled implementation of all of them in the light of current neural modelling research, particularly as discussed by Koch [1999], who brings together work on neuroscience with models used for neurons in analog VLSI research. We expect this to ease potential future collaborative work with researchers from both fields.

2.2 Description of the model

The overall behaviour of the neurons is similar to single compartment "leaky integrate and fire" models. Their state is a representation of the potential difference across the neural membrane (the membrane potential). They have a base potential, which they will return to in the absence of external input, and synapses attach to them and raise and lower their potential when their pre-synaptic neurons spike. If the potential rises above a specific level, the neuron will "fire", sending a spike to any connected output synapses; then the potential will reset to some lower value, and the neuron will enter a "refractory period" when it will be unable to receive synaptic input.

The synapses come in a variety of forms. All are adaptive with short-term facilitation and depression (with variable size and time constant). This allows the synapses to affect the post-synaptic neuron more or less strongly depending on whether they have recently received other spikes. Indeed, it is possible to combine the effects so that, for instance, the synapses will respond increasingly strongly for the first few spikes, and then that response will die away to nothing until the synapse has had a period of time to recover. They also all have variable synaptic delays (the time from the firing of the presynaptic neuron to the reception of the excitation or inhibition by the post synaptic neuron). In the real world this delay is actually a combination of axonal, synaptic and dendritic delays. These are all combined as a single variable in the model.

Some of the model synapses are voltage dependent, which is to say they try to pull the neural state to a level fixed for each synapse. This is biologically realistic and models the battery potential of the ion channels which open when a synapse is activated, which then pull the membrane potential towards that level. Others are current injection synapses which deposit a small or large packet of charge into the neuron, thus directly raising or lowering the membrane potential. This synaptic model more closely resembles a typical ANN neuron but is less biologically realistic. It is useful in some circumstances however, particularly for non-spiking inputs, *e.g.* from sensors, where the size of the packet of charge deposited can be proportional to the input.

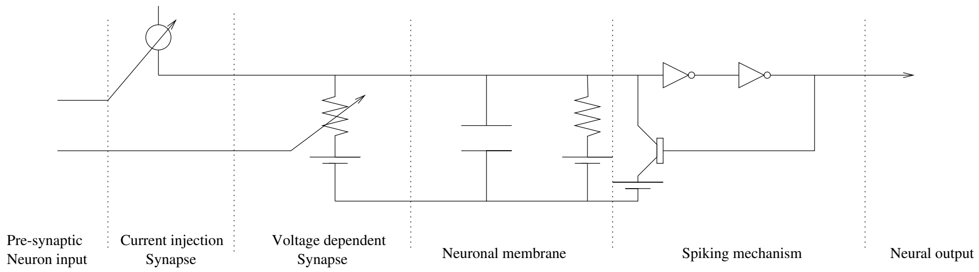

Figure 1: A simplified electrical model of a neuron with synapses

The models described by Koch (*ibid.*) have many of these features, and he explains them in terms of their electrical properties, thus creating neurons and synapses based on electrical circuits (this makes it very suitable for VLSI implementation). The features described above are given in the simplified circuit shown in Figure 1. The neuron is an RC circuit with a fixed (membrane) capacitance, and basal potential and conductivity, to which the neuron will decay exponentially at a rate determined by the capacitance and conductance in the absence of external input. The two types of synapse also have electrical implementations. The first, voltage dependent, type is modelled as an additional (variable) conductance and battery, which will pull the output neuron membrane potential towards the potential of the battery with a strength determined by the conductance — this decays exponentially to zero, but receives a boost when a spike arrives from the input neuron. The second, current injection, synapse is simply a current source which deposits a packet of charge on the output neuron over a short period of time for every spike received; this will increase or decrease the membrane potential of the neuron directly. The size of the boost to the voltage dependent synapse conductance, and the size of the current source supplied by the current injection are effectively the 'weights' of the connections. These values are affected by the facilitation and depression mechanisms described above (not shown in the figure).

2.3 A neural circuit for song recognition

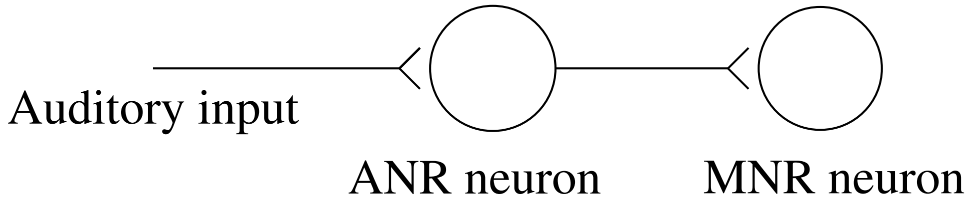

Figure 2: A simple network of two neurons

A small collection of these neurons and synapses is capable of quite sophisticated behaviours. For example the two neuron model shown in Figure 2 can detect cricket song, requiring both the correct syllable length, the correct inter-syllable gap, and the correct number of syllables for the second neuron to spike. An example of the behaviour is shown in Figure 3.

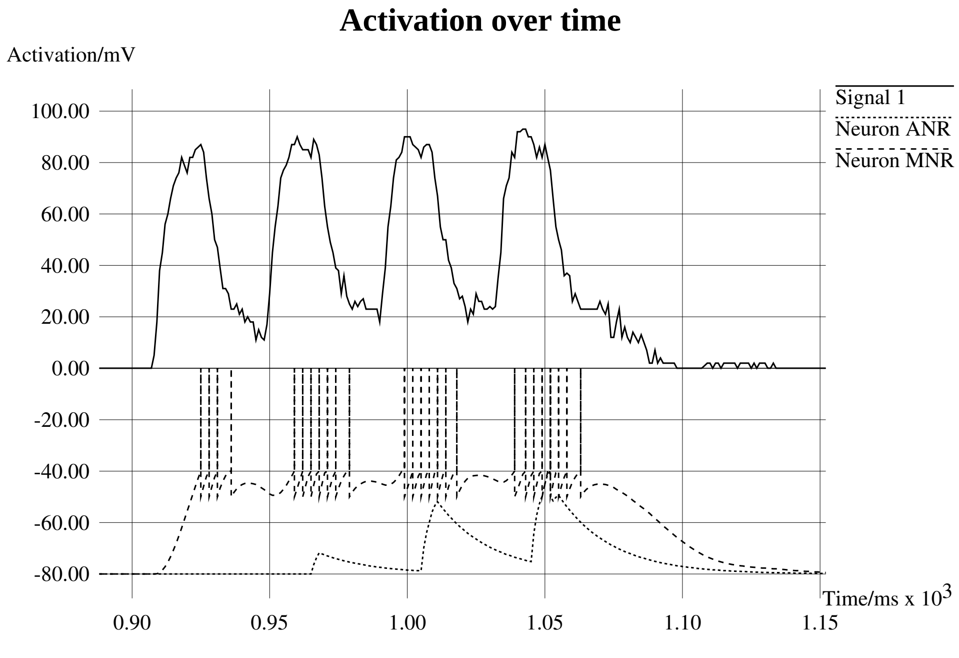

Figure 3: Response of simple network to appropriate input

The synapse connecting to the first (ANR) neuron is an excitatory current injection synapse. It has low initial strength, but has facilitation which quickly allows it to spike frequently on receiving a series of large inputs. The synapse between the two neurons is more complex. It has no initial strength, so the first spikes have no effect, and also has high depression on the synapse with a short time constant which quickly inhibits the synapse completely. However, there is also a small but long time constant facilitation, which slowly builds during each syllable, and whereas the depression decays away almost completely between syllables, the facilitation only loses a small amount of its power, and so by the fourth syllable it is strong enough to briefly overwhelm the depression and excite the neuron enough to produce a single spike. With a simpler neural model, many more neurons would have been necessary to achieve this task.

2.4 A neural circuit for phonotaxis

The full neural model used to control the behaviour of the robot in the experiments shown below is shown in Figure 4. A two neuron system similar to that described above is implemented for each ear, and a cross-inhibitory connection between them ensures that the only the stronger auditory signal will excite a response. The “MN” neurons are connected by current injection synapses to the “motor” units, which are non-spiking neurons that integrate the incoming potential and drive the motors at a speed corresponding to their current membrane potential. Thus, for the correct song pattern, the robot will tend to turn towards the side where the song is louder. The default forward speed is controlled by a tonically firing neuron that excites both motor units.

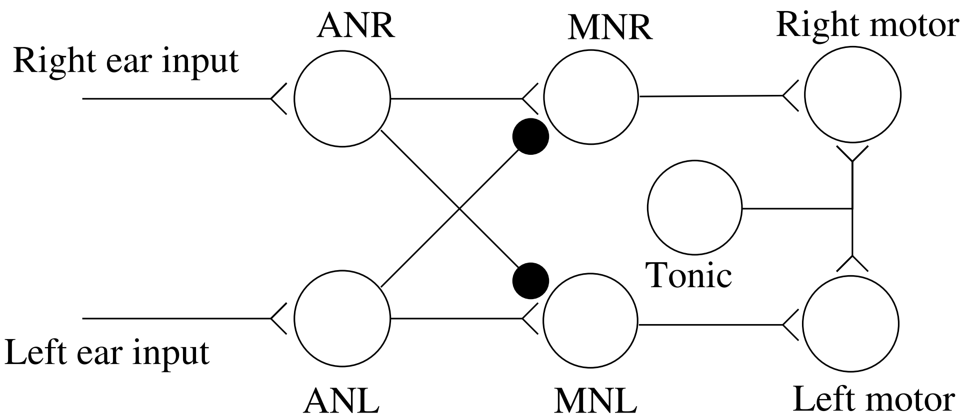

Figure 4: Network used in experiments

3 Hardware

The robot is a Koala^{TM}. The chassis is approximately $30 * 30 * 20$ cm, and has three tyred wheels on each side driven by a pair of DC motors; it is intended to be capable of movement on outdoor terrain. The processor is a Motorola 68331 @ 22 MHz; and it is programmable in C (and assembler). The auditory processing was carried out by custom built analog circuits designed to mimic the animal’s sensors, interfaced to the processor through A-D ports. This auditory circuit is described in detail by Lund et al. [1997]. Two microphones separated by 18 mm (1/4 wavelength of the carrier frequency — 4.7 kHz — of cricket song) receive and amplify the sound. The signal from the left microphone is delayed by 53 μs (1/4 the phase of 4.7 kHz) and then subtracted from the right; and vice-versa. The delay and relative weighting of the two signals is programmable. The amplitude of the resulting waves is measured using a RMS circuit, resulting in two analog signals representing the amplitude of the two ears.

The sensor readings from the robot are transmitted to the offboard computer (via the serial port) at a rate of approximately 1 kHz. The raw signals are scaled for input to the auditory neurons. The output from the motor units is transmitted back to the robot at a rate of 10 Hz and used to set the motor speeds.

The auditory stimulus was a simulated cricket song. A single “syllable” of the male cricket song is a 20 ms burst of almost pure 4.7 kHz sine wave. The syllables occur in groups of 4, with intersyllable gaps of 20 ms; and the groups (“chirps”) repeat at approximately 3Hz. This sound could also be simulated using a customized program under Linux, writing directly to the sound card. Variants that differed from this “ideal” song were also generated and tested for comparison, in particular continuous sound and faster and slower syllable rates (which the cricket ignores) and different syllable lengths at the same syllable rate (which the cricket will approach).

4 Experimental Results

Initial experiments were done with the network shown in Figure 2 without the robot. Two sources of input were used, the first and simplest was a square wave of the correct frequency; the second was recorded from the ears on the robot itself. Once parameters had been found which allowed selection of the correct syllable structure, the full network shown in Figure 4 was used. Finally, after tuning the cross inhibition and the tonic neuron, experiments were carried out on the robot, and the motor response itself was calibrated.

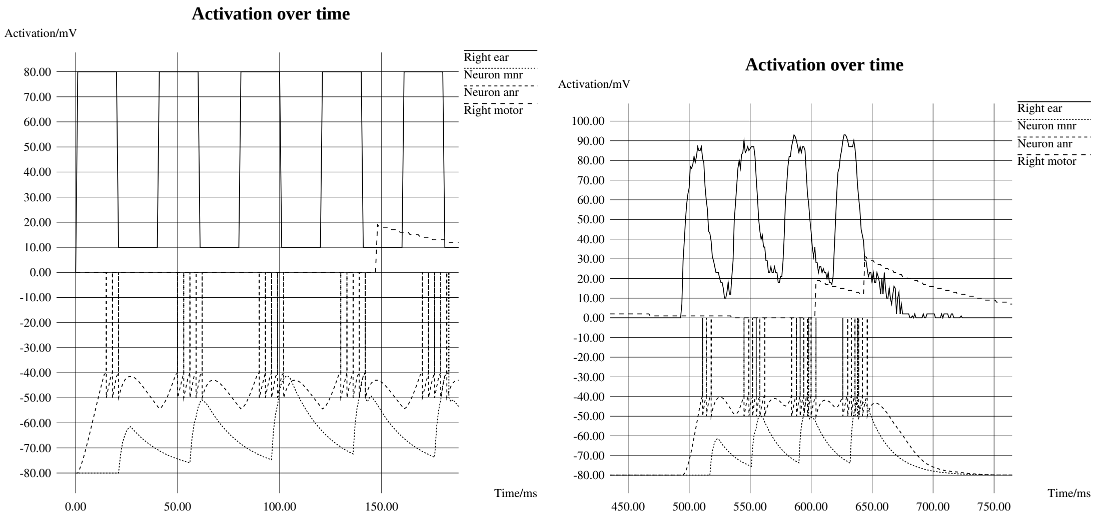

Figure 5: Network detects correct syllable length: left simulated input, right real sound

The parameters of the simple network are changed slightly from Section 2.3. That network could distinguish different syllable lengths and gaps, but a continuous tone and random noise both triggered the MNR neuron. The synapse between the two neurons was given a low initial strength, and the time constant of the depression was increased to decrease the long term effect of continuous spiking. The results of this with simulated and recorded inputs can be seen in Figure 5 — the simulated song produces one spike per chirp in the fourth syllable, and the recorded signal produces one each in the third and fourth syllables. The motor output caused by these spikes is also shown in the figure.

Figure 6 shows what happens when the syllable duration and gap are increased or decreased too far — the MNR neuron is stimulated, but not enough to allow it to spike, as can be seen from the flat motor output. Likewise Figure 7 shows the effect of a pure tone and random noise on the network. Again the MNR neuron is stimulated, but it does not spike with a constant input, and only very rarely with a random input (not at all in this time sequence).

Cross inhibition was then added to allow the network to control phonotaxis in the robot. Without it, the MN neurons simply spike more and more the closer they get to a sound source driving the motors faster and faster. Synapses with shunting inhibition[1] were added from the AN neurons to the MN neurons on the opposite side. This allows competition between the sides, with the stronger AN neuron (on the side closer to the sound source) overriding the weaker, but the latter also moderates the amplitude of the other MN neuron somewhat. In the example shown in Figure 8 the left ear input is only 10% weaker than the right, which is much less than is often found, but this is a sufficient difference for the cross inhibition to allow only the MNR neuron to fire.

A tonically firing neuron is also present to give a base speed to the motors, so that the default behaviour is to drive forward at a fixed speed. Figure 9 shows the course of the robot tracking a simulated cricket song. Experiments on the robot are currently being carried out, the results of which will be presented at the conference.

[1]Synapses whose battery potential is the same as the output cell membrane's base potential

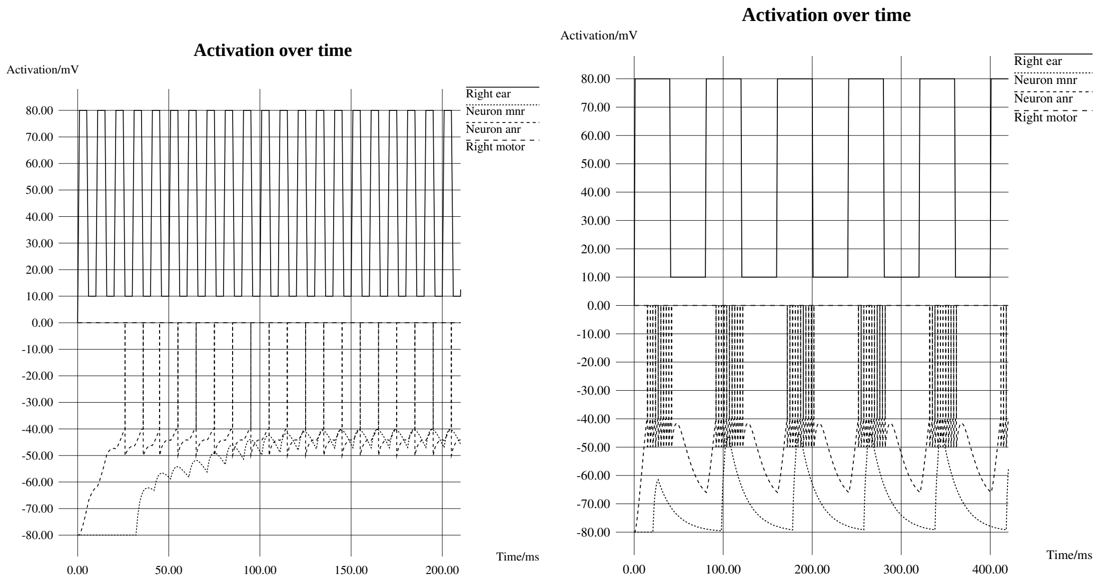

Figure 6: High and low frequency syllables are rejected - MN trace does not reach threshold, motor stays at 0

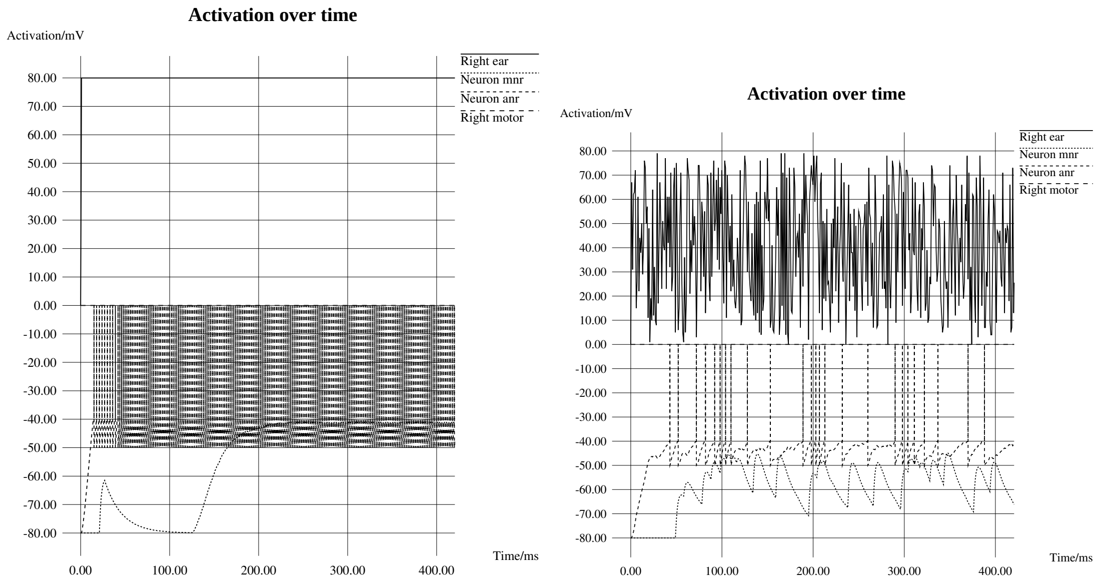

Figure 7: Continuous sound (left) and random sound (right) are also generally rejected

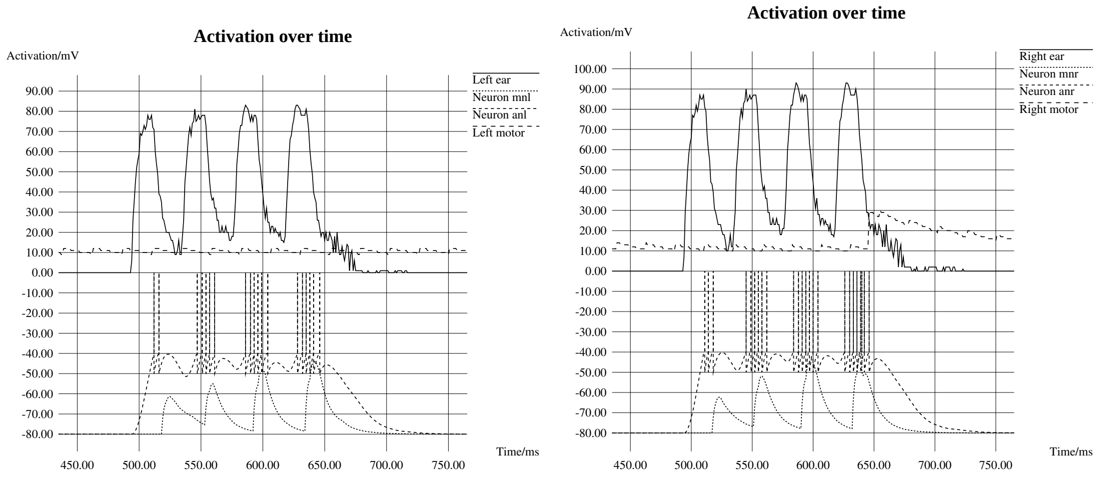

Figure 8: Cross inhibition allows direction selection: When the left ear (left plot) has slightly lower amplitude than the right ear (right plot) only the right motor is activated.

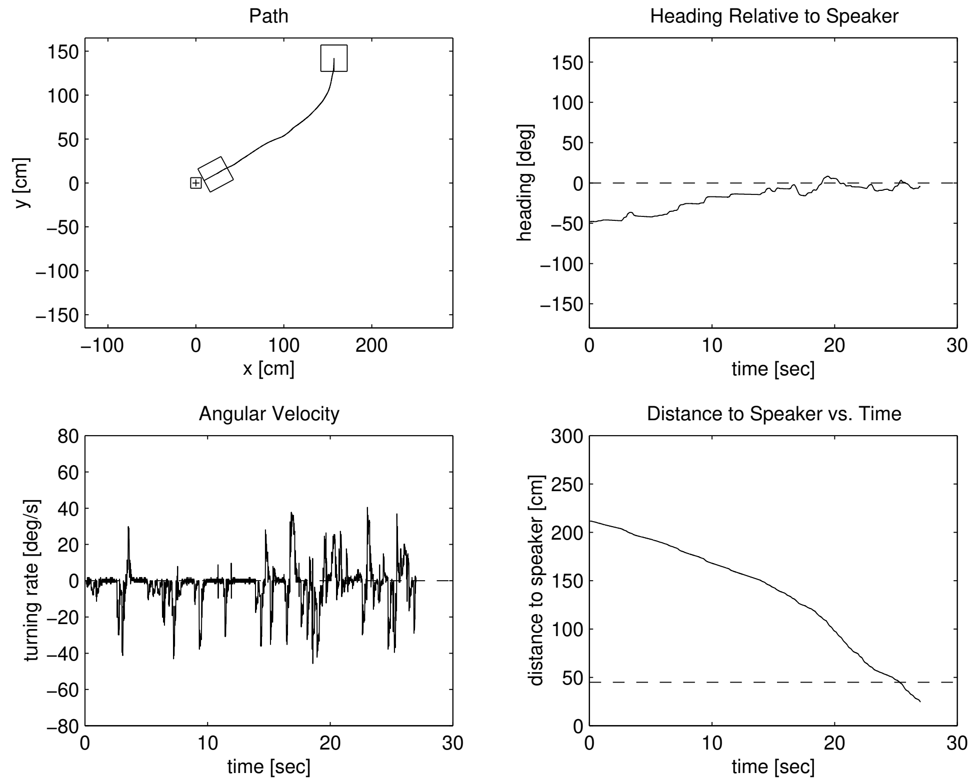

Figure 9: Course of robot tracking cricket song

5 Discussion

The approach to "biologically-inspired" robotics that we have adopted assumes that to exploit biological designs we need to understand what biological systems are actually doing. Thus we have opted for close study of specific systems, and aimed to reproduce the behavioural capability in some detail, as well as demonstrate that the system matches the known neurophysiology. Given that few biological systems are well enough understood to directly reimplement them in an engineered system, this approach necessarily becomes as much a contribution to biological hypothesis building and testing as it is to robotics. Indeed it could be said that the "application" for which we are currently building robots is "biological modelling".

The neural model introduced in this paper is intended to be a generic tool for use on such robots. It has a number of characteristics that distinguish it from other models (particularly from standard neural nets) in ways we believe to be important in understanding the flexibility and adaptivity of animal behaviour. In particular it represents some of the interesting dynamics of synaptic adaptivity rather than use the simple, instaneous "weight" model common to both ANNs and many "integrate-and-fire" implementations. The result is that it easier to implement various behaviourally important characteristics. This is not simply a case of a more complex individual neural model trading off with number of neurons needed to produce a particular function; but that there is a natural match between the neural mechanisms and the desired behavioural characteristics. This is illustrated in our results by how, with a few neural connections, it is possible to reproduce the preference patterns shown by crickets in tracking songs. While this is to some extent a replication of previous work, the current model is an advance for several reasons. First, there are fewer arbitrary assumptions in the modelling of the low-level neural mechanisms, and more accurate approximations of the neural dynamics have been used. Second, the model is in fact more effective in copying all the details of the cricket's response, e.g. the preference for syllable rate independent of duty cycle. Finally, the planned mapping to motor output is implemented in a more biologically plausible manner: this aspect needs further attention in future work, particularly when we come to use the model on a legged robot.

There are several other issues to be addressed. One obvious problem is that of parameter tuning as the complexity of the model increases. Using biological details as the constraint is the obvious approach to take here but the data is generally not available in exactly the form required. What we hope to develop is some familiarity with and a "library" of ways to achieve useful functional "mini-circuits" that can act as building blocks in more complex designs.

The aim in the immediate future is to use the model for a neural implementation of the integration into phonotaxis of the optomotor reflex (which so far we have only been done algorithmically). This reflex normally acts to stabilise the animal relative to its visual surround by compensating for rotational velocity with appropriate torque. When the animal or robot is intentionally rotating it needs to ensure that the generated visual velocity, in this case, does not lead to an unwanted correction. At least five possible schemes for doing so can be found in the biological literature, including simple inhibition and efferent copy. As Collett [1980] has pointed out, some of these algorithms appear identical at one level but they differ fundamentally when the actual dynamics of the sensory response and motor control are considered. What we wish to investigate is how they differ with respect to neural implementation: are some schemes substantially more plausible for insect brains than others? Can specific predictions be made that would allow us to determine which mechanism is actually used by the animal? And might these mechanisms suggest novel solutions to the problem of sensory integration in robots?

References

Christopher Bishop. *Neural Networks for Pattern Recognition.* Oxford University Press, 1995.

T. Chapman. *Morphological and neural modelling of the orthopteran escape response.* PhD thesis, University of Stirling, Stirling, 2001.

T. Collett. Angular tracking and the optomotor response: an analysis of visual interaction in a hoverfly. *Journal of Comparative Physiology A*, 140:145–158, 1980.

G. Indiveri. Modelling selective attention using a neuromorphic analog VLSI device. *Neural Computation, (in press)*, 2000.

C. Koch. *Biophysics of computation.* Oxford University Press, 1999.

H. H. Lund, B. Webb, and J. Hallam. A robot attracted to the cricket species *gryllus bimaculatus.* In *Proceedings of 4th European Conference on Artificial Life.* MIT Press/Bradford Books, MA., 1997.

D.E. Rumelhart and J.L. McClelland. *Parallel Distributed Programming*, volume 1 and 2. MIT Press, 1986.

B. Webb. A robot cricket. *Scientific American*, 275(6):94–99, 1996.

B. Webb and R. Harrison. Integrating sensorimotor systems in a robot model of cricket behaviour. In *Proceedings of SPIE*, 2000.

B. Webb and T. Scutt. A simple latency dependent spiking neuron model of cricket phonotaxis. *Biological Cybernetics*, 82(3):247–269, 2000.