Design and development of model robots: a new dimension in the delivery of electronic and computer engineering programs

Girija Chetty
Gippsland School of Engineering, Monash University, Australia
E-mail: Girija.Chetty@eng.monash.edu.au

Abstract This paper reports two senior-level design projects in electronic and computer engineering study programs. The projects involve the design, development and construction of model robots using embedded controllers and microprocessors. The teaching technique not only motivates the students to develop a thorough understanding of the embedded controllers and microprocessors, but also helps them in the development cycle and to understand useful and innovative engineering applications. Details of the model robots and learning outcomes of the methodology used are presented.

Keywords electronic and computer engineering; embedded controllers; model robots

Gippsland School of Engineering at Monash University delivers engineering courses in various disciplines including electronic and computer engineering, electromechanical engineering, and civil and mechanical engineering programs in on-campus and distance education (off-campus) modes. In the electronic and computer engineering program, students are academically trained to work as professional electronic and computer engineers. Particular areas of specialisation include analog electronics, digital electronics and computer hardware including computer systems engineering, communications and networking, and computer software including operating systems and software engineering. The digital systems stream in the degree includes several units in digital electronics, computer systems engineering, digital signal processing and the study of EDA tools.

The design, development and construction of model robots using embedded controllers and microprocessors provides a new and more effective technique for delivery of digital systems units in electronic and computer engineering programs. The teaching technique not only motivates the students to develop a thorough understanding of the embedded controllers and microprocessors, but also helps them in the development of a useful and innovative application. In this paper details of two such senior design projects are presented.

The first project involves the development of a voice-controlled mobile robot using a speech recognition card and M68HC11 microcontroller-based mobile platform and the second project is about the development of a hexapod robot using a BASIC STAMP (PIC-based microcontroller). The model robot projects allowed students to grasp the principles of computer systems engineering (embedded controllers and microcontrollers) and techniques of developing a useful application by integrating various sub-systems. The next section describes the design and development of a voice-controlled robot, followed by a section describing the development of a hexapod based on biological locomotion principles. The paper concludes with the outcomes of the methodology used and future plans.

Model robot project I

This robot responds to simple voice commands and executes a task. The control scheme for the robot was based on a hybrid control scheme with reactive or behaviour-based architecture. The hardware implementation involved a M68HC11 microcontroller and a HM2007 voice recognition chip. Such a robot can serve as a model in understanding the development of an aid for speech-impaired individuals. The goal for the development of such a robot was to create a device that would allow a person to function at the same level as a person who does not have a disability. There are two aspects of a model robot that affect its functionality: the interface between the user and the robot, and the interface between the robot and the objects it is manipulating.

Typical interfaces between the user and the robot are keypads, joysticks, and switches. A user interface of this type has limited use particularly for individuals with hand motor impairment, or tremor as observed in patients suffering from multiple sclerosis or Parkinson's disease. In such cases, speech as an interface tool between the user and the model robot will be more useful. The interface between the robot and the objects being manipulated is the hand of the robot, also referred to as the end effector, consisting of tools that are matched to the tasks being performed. For example, a robot equipped with a gripper will be able to perform 'pick and place' activities. Although limited, a large portion of the activities we perform in our daily lives involve 'pick and place' tasks. Adding mobility to the robot, in addition to the object manipulation task, will increase the functionality of the rehabilitation robot. The development of a speech recognition circuit is described here.

Speech interface circuit

Most of the time, we take our speech-recognition abilities for granted. Present-day speech recognition systems cannot distinguish when a person speaks among several in a party. Speech recognition in general is classified into two categories: speaker independent and speaker dependent. Speaker-independent products can recognise speakers with different pitch, accent or both. But they have the disadvantage of handling a smaller vocabulary for recognition. The speaker-dependent systems, on the other hand, are trained by the individual who will be using the system. These systems are capable of achieving a high command count and better than 95% accuracy for word recognition. The drawback to this approach is that the system responds accurately only to the individual who trained the system. For individuals with hand-motor disabilities, a speaker-independent system can be used, but a speaker-dependent system is necessary for individuals with verbal impairment such as cerebral palsy patients. By training the system it can be used by individuals with other types of verbal impairments.[1–8]

Speech recognition systems have another limitation concerning the style of speech they can recognise. They can recognise three styles of speech: isolated, connected and continuous. Isolated speech recognition systems can handle words that are spoken separately, where the user has to pause between each word or command spoken. Connected speech recognition is a halfway point between isolated word and continuous speech recognition. Connected speech allows users to speak multiple words. Continuous speech is the natural, conversational speech we are used to hearing in everyday life. It is extremely difficult for a recogniser to shift through the text, as the words tend to merge together. There are several speech recognition products available on the market which can do speaker-dependent, speaker-independent, isolated, connected or continuous voice recognition. We selected a HM2007 voice recognition chip and configured the chip to identify words or phrases that are 1.92 seconds in length, in speaker-dependent mode. The 1.92 second word length option reduces the word-recognition dictionary number to 20. The picture of the speech recognition circuit is as shown in Fig. 1(a).

The circuit allows a HM2007 voice recognition chip to be trained with a keypad and a microphone. We trained the circuit by pressing '1' and then 'T' (training) for the first word. If the circuit accepts the input as the first word, the diagnostic LED on the board flashes. Training can be continued for all 20 words by pressing the appropriate key, followed by the 'T' training key on the keypad. When a trained word is spoken the circuit recognises the word by raising the appropriate BCD output high. We reserved four word spaces for each command, allowing the robot to recognise five commands. The BCD outputs can then be used to interface with the robot controller.

Robot controller implementation

Early robot control systems attempted to utilise their knowledge of the world to plan actions for situations. Robotic researchers created robot control systems by reasoning about and planning each action. Because of the limited successes of these systems based on planned architecture in the real world, attempts were made to eliminate the requirements for world knowledge altogether through the use of so-called reactive or behaviour-based systems. The basic idea underlying reactive control systems is the idea of 'behaviour'. In contrast to planning systems whereby the control architecture is split into functional tasks (sense, model world, plan, execute), reactive systems are built as multiple independent tasks which operate in parallel. Each behaviour processes its own sensory information and issues its own motor commands. In order to coordinate the final motor commands, each behaviour can disable those other behaviours which are in conflict with itself. When introduced, reactive control systems demonstrated navigational capabilities that were quicker and better than those of planned systems. We used a similar behaviour-based architecture for

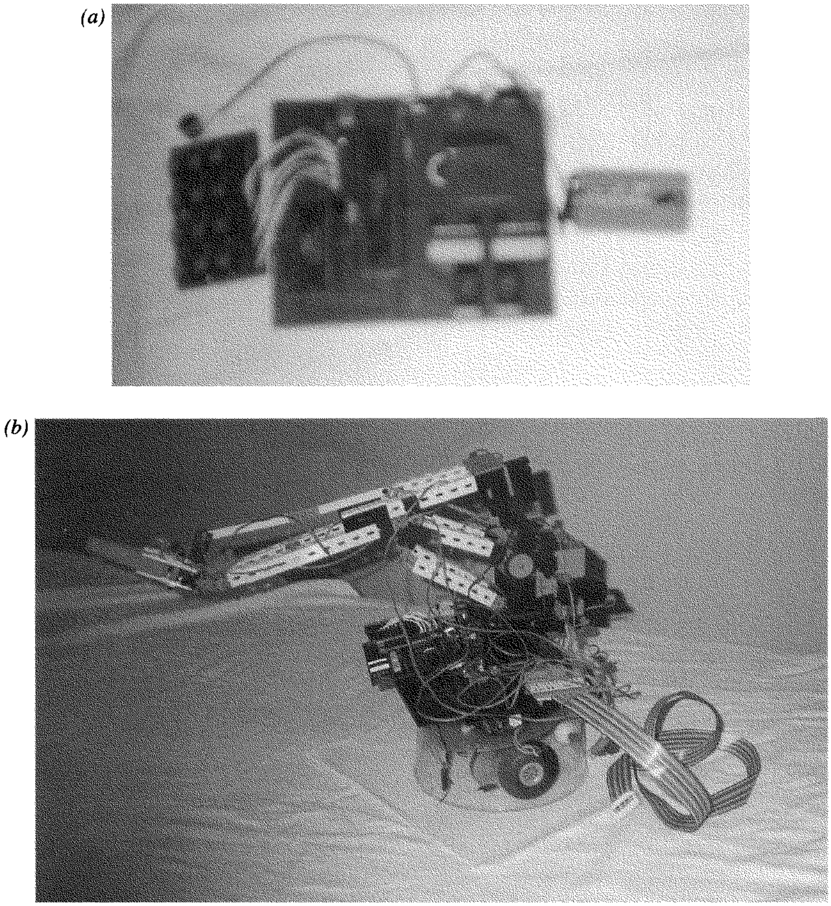

Fig. 1 *(a) Speech recognition circuit; (b) model robot.*

controller implementation. The block schematic for the controller architecture is as shown in Fig 2.

The voice-controlled mobile robot with control architecture as described above was implemented as a model robot. The model vehicle comprised three autonomous sub-systems. The first sub-system was the voice-controlled user interface using a HM2007 voice recognition chip.[12] The second sub-system was mobile robot with controller, a RugWarrior brains and brawns kit,[9,10] and the Fishertechniek manipulator kit as the third sub-system.[11] The model robot accommodates a speech recognition board, all the obstacle avoidance sensors and actuators for a mobile unit, the 3 degrees of freedom manipulator and the controller. The robot was programmed to service each sub-system in predetermined sequence. Each autonomous sub-system communicates with other sub-systems through appropriate communication protocols. Figure 1(b) shows the model robot with associated sub-systems. The hardware and programming details of the sub-systems are given in Refs. 9–12.

Model robot project II

This project is about the design and development of a model hexapod robot. Walkers are a new class of robots that imitate the locomotion of animals and insects, and use legs for locomotion. Locomotion by legs is hundreds of millions of years old. In contrast, wheels are recent, but are more popular and require a relatively smooth surface to ride upon. Walkers have the potential to traverse through rough terrain that cannot be traversed by wheeled vehicles. With appropriate control of leg movements, a legged robot can climb steps, cross ditches, and walk on extremely rough terrain, where the use of wheels would not be feasible. However, an important drawback of legged locomotion, compared with wheeled locomotion, is the much higher complexity involved in its control, even in the case of a completely flat background.

Controlling walking machines has always been a challenge for roboticists because of the large number of degrees of freedom. Due to the large number of degrees of freedom, and the complexity of legged locomotion, human real-time control of individual joint or leg movements is almost impossible in practice.[11] This means that a walking machine, even if it is human driven, must show autonomous behaviour at least at the levels of joint actuation and leg coordination, providing automatic terrain adaptation and body stabilisation. Until now, the lack of reliable and efficient algorithms for adaptive walk control on difficult terrain has made the use of legged locomotion impractical for many applications that, in principle, could benefit from it. The main reason is that because of the large number of degrees of freedom, control of legs cannot be

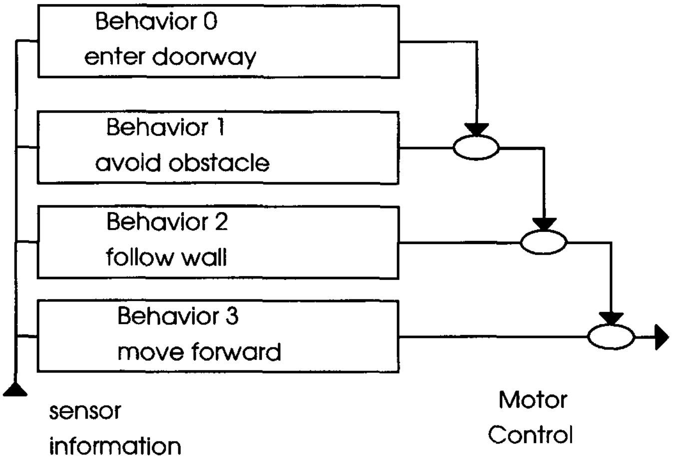

Fig. 2 *Behaviour-based robot control architecture.*

treated independent of each other, e.g. the movement of a single leg has to be considered in the context of all other legs.

Since the capability to deal with difficult terrain is the key feature that gives interest to legged robots, it makes little sense to develop walking controllers under the assumptions of smooth terrain. On the contrary, the presence of arbitrary irregularities in the ground should be considered as the typical situation, in which obstacles of any size, including walls and cliffs, may appear anywhere.[12] Thus, it is hard or even sometimes impossible to find a simple, flexible and direct programmable strategy, enabling the walking machine to adapt to a changing environment and rough terrain.

Several kinds of control structure have been presented to address this issue.[13–16] One author[13] proposed a kinematic model in a simulation based on a reference trajectory. The reference trajectory was divided into four segments, thus offering the walking machine multiple moving patterns when the interior or exterior conditions changed. Another researcher[14] proposed a control structure for the locomotion of legged robots under the assumption of difficult terrain from the very beginning. The structure considered aspects of stability, mobility, ground accommodation, gait generation, and robot heading, integrated in a coherent way by three-level hierarchical task decomposition. The control structure was implemented on Ghengis II, a commercially available six-legged robot, as behaviour-based control modules, according to the main guidelines of subsumption architecture. Even though the implementation aspect appears simpler, the control strategy, being complex, assumes highly structured terrain and cannot adapt to unexpected terrain conditions.

Features of neural networks include fault tolerance, the ability for processing noisy sensor data and the possibility to learn from examples or according to given evaluation functions. Neural networks proved to be more appropriate tools for implementing adaptive controllers, enabling the walking machine to adapt to changing environments. Several researchers proposed control structures based on neural networks for legged locomotion. Most of the techniques were based on reinforcement learning (RL), because of their potential for self-supervised learning of walking behaviour and on-line adaptability to changing environments.[13–16] These approaches use different types of RL algorithms (AHC, Q-learning), and different network representations (CMAC, RBF networks, Backpropagation). With these RL-based structures, it is possible to learn leg trajectories represented as sequences of joint angles over time. However, the learning process has to be accelerated in order to allow on-line adaptation during the execution of walking behaviour in unstructured terrain.

However, using RL-based approaches for complex technical applications is extremely difficult because of the huge number of training cycles needed to learn optimised control strategies and the difficulty of ensuring the safety of the robot during the learning process. Thus it is practically very difficult to make RL-based algorithms work in real robots because the actions of real robots operate under real-time constraints. Due to the real-time constraints, the behaviour of the system changes while the control rule has to be learned, a problem with which very few algorithms can cope, or the actual performance time ends before the learning algorithm was able to learn anything.

Thus, despite the advances in issues related to the mechanical design of legged robots and control and coordination of legs during locomotion, the performance of current legged robots remains far below even their most simple counterparts in the biological world. Naturally, this has led to a search by researchers for biologically motivated locomotion principles.

Biological locomotion principles

Animals have evolved to occupy every environmental niche where we might want an autonomous robot to operate. As a result, they provide proven solutions to the problems of navigation, locomotion and sensing, often in the most difficult of environments.[17,18] In crustaceans, many action patterns such as these underlie locomotion and feeding can be evoked by stimulation of single neurons or sets of neurons.[17,18] Thus, there is a correspondence between units of behaviour, their modulation and underlying neuronal components. In the development of this project, a biologically motivated control architecture based on these neuronal components and modulatory principles was used, instead of computational intensive forward and inverse kinematic solutions. The architecture is based on state sequences that are derived from analysing the animal behaviour.

The neuronal mechanisms underlying locomotion were initially established by a study of simple animals including lobsters and crabs, insects, sea slugs and worms.[17] Experiments conducted on these animals show evidence of central pattern generators (CPGs), that generate patterns which resemble limit cycle oscillations. These patterns are often invoked using a small number of neural pathways by higher level processes, or afferent sensory inputs, or both. In the case of locomotion, these patterns appear to encode particular phase and frequency relationships between the movement of legs. Inspired by this, many researchers have proposed a number of multidimensional oscillator models for locomotion where each dimension would produce the patterns necessary to drive a single leg of the system.

The fundamental governing concept of the CPG model is to couple various oscillator models to produce spatio-temporal symmetries that correspond with characteristic gaits observed in biological systems. Thus the gait generation problem reduces to the following:

- Generation of a single pattern generator or oscillator
- Connecting the output of the oscillator to appropriate time delays
- Coupling the outputs of the time delay elements and the original oscillator output with appropriate weights for gait generation

Figure 3(a) shows a block schematic for gait generation from oscillator output and time delay elements, where $Y1(t)$, $Y2(t), \ldots, Yn(t)$ is the gait pattern derived from CPG oscillator output $X(t)$ and time delay blocks.

The CPG-based gait model as shown above can be derived from first principles. The dynamic model of CPG, $x(t)$ can be represented as follows:

Patterns of y $(y1, y2, \ldots, y6)$ may be considered to represent the motions required from each leg of the robot.

$$\begin{aligned} &y(t) = \hat{p}\sin wt \text{ as } t \to \infty \\ &\dot{x}_1 = x_2 \\ &\dot{x}_2 = -a_1 x_1 - a_1 x_2 + p\cos wt \end{aligned} \tag{1}$$

and the output $y = x1$.

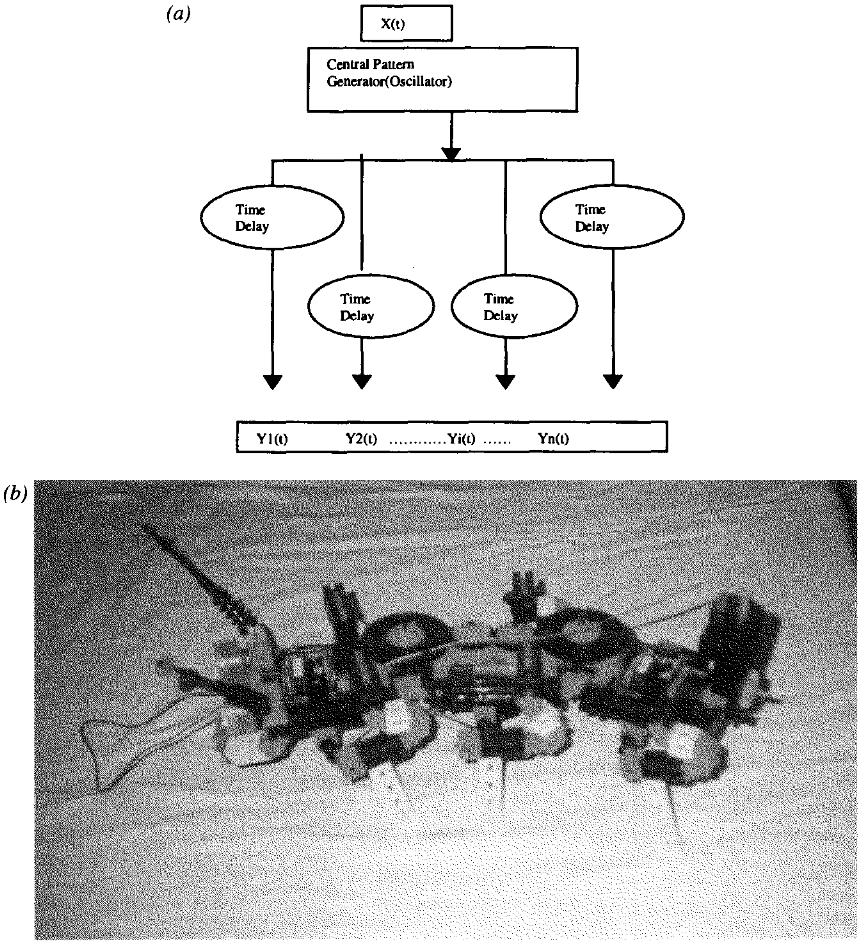

Fig. 3 *(a) Basic gait generation model; (b) model hexapod robot.*

Equation (1) would be stable if $a_1, a_2 > 0$. Also for certain values of a_1, a_2, the system converges to steady state oscillations. Equation (1) with appropriate values of a_1, a_2 is referred to as a pattern generator.

Now the time delay block needs to be generated for gait generation. A pure time delay element can be modelled using a transfer function of the form e^{-Ts}, where s is the Laplace variable.

Such a transfer function can be realised using infinite dimensional filters or Pade approximated finite dimensional filters. The gait model can be made robust to noise and external disturbances by adding nonlinearity of a specific type such as the van der Pol type of nonlinearity. This modifies eqn (1) to the form:

$$\begin{aligned} \dot{x}_1 &= a_1 x_2 \\ \dot{x}_2 &= -a_2(\chi_1^2 - a_3)x_2 - a_4 x_1 - a_5 x_2 + p\cos wt \end{aligned} \tag{2}$$

The CPG-based gait generation was implemented on a BASIC STAMP II based microcontroller kit. Model robot implementation based on central pattern generators for gait realisation was done by integrating a Lynxmotion hexapod kit and BASIC STAMP II for tripod gait. The robot walks using the alternating tripod gait. For a stable tripod gait, legs 1, 4 and 5 or 2, 6 and 3 forming a tripod should be on the ground. Each of the six legs had two degrees of freedom, rotation and extension, and whisker sensors. The robot's six legs are controlled by three servos to provide full motion and over 2″ of vertical leg lift. A host PC is required to download programs to BASIC STAMP II.

The robot was programmed to walk forward, reverse and turn left or right. Figure 3(b) shows the model hexapod robot.

Results and conclusions

The design and development of model robots reported in this paper allowed us to achieve various learning objectives such as:

- Study of different types of computer-based controllers, including embedded controllers, M68HC11 and BASIC STAMPs
- Visualisation and development of a useful application such as a robotic aid for the speech impaired and finding a solution to a complex terrain problem using biological locomotion principles
- The techniques for development of a rapid prototype of the robot design by integrating various off-the-shelf sub-systems instead of developing the system from scratch
- Making the study of computer systems engineering a motivating and interesting experience by development of such model robots
- Increased teamwork and project management skills, which is an essential requirement in training the engineers for the new millennium.

The student feedback on the projects implemented is very promising as it allows the students to play with model robots and combine learning with playing. Also, the approach being different from conventional electronic and computer engineering program delivery, it attracted students from other disciplines to undertake similar projects. Further plans involve extending and exploring several such new techniques and making it suitable for distance education delivery.

References

1 Stephen Edelson, 'Multiple Sclerosis', Environmental and Preventive Health Centre of Atlanta Web report, *http://www.ephca.com*, pp. 1–3 (1995).

2 Applied Technologies Division, 'Introduction to rehabilitation robotics', Rehabilitation Technologies Division, Web report, *http://www.appliedresource.com/RTD/RehabInt.html*, pp. 1–4 (1998).

3 Rodney Brooks, *A robust layered control system for a mobile robot*, MIT A.I. Memo 864 (Massachusetts Institute of Technology, 1985).

4 Arthur Ed LeBouthillier, 'Trends in robot control architectures', Web article, *http://www.cyberg8t/pendragn/trends.htm*, pp. 1–3 (1998).

5 F. R. Noriels and R. G. Chatila, *Control of mobile robot actions, Proc. IEEE Int. Conf. on Robotics and Automation (ICRA-89)*, Arizona, 1989, pp. 701–707.

6 Jorg-Michael Hasemann, 'A control architecture for service robots', Web article, *http://www.ele.vtt.fi/aut/gwics95.htm*, pp. 1–6 (1998).

7 Kai-Fu Lee, *Automatic Speech Recognition* (Kluwer, Amsterdam, 1989).

8 Hasan Abu-Zaina, David Kirk and A. Almer, *A speech recognition system for operation of a visual inspection workstation*, Hoyer Institute for rehabilitation research and service, Wichita State University Web report, RESNA'96 proceedings, pp. 43–146, 1998.

9 *RugWarrior Mobile Robot kit User's guide* (A. K. Peters Publishers, 1993).

10 *Fishertechniek Profi Computing Kit User's Manual* (1995).

11 F. Kirchner. 'Q-learning of complex behaviours on a six-legged walking machine', *Robotics Autonomous Syst.*, **25** (1998), 252–262.

12 Enric Celaya and Joseph Porta, 'A control structure for the locomotion of a legged robot on difficult terrain', *IEEE Robotics and Automation Magazine*, June 1998, pp. 43–51.

13 J. C. Gallagher *et. al.*, 'Application of evolved locomotion controllers to a hexapod robot', *Robotics Autonomous Syst.*, **19** (1996), 95–103.

14 K. Iganaki, 'A robot walking on rough terrain with reflector motion', in *Proc. 5th Int. Conf. on Control, Automation, Robotics and Vision (ICARV'98)*, Singapore, 9th–11th Dec. 1998, pp. 469–473.

15 W. Ilg and Berns Karsten, 'A learning architecture based on reinforcement learning for adaptive control of the walking machine LAURON'. *Robotics Autonomous Syst.*, (1995), 321–334.

16 W. Ilg *et. al.*, 'Hybrid learning concepts based on self-organising neural networks for adaptive control of walking machines', *Robotics Autonomous Syst.*, **22** (1997), 317–327.

17 J. Z. Ayers *et. al.*, 'A modular behavioral-based architecture for biomimetic autonomous underwater robots', in *Proc. Autonomous Vehicles in Mine Countermeasures Symposium*, Naval Postgraduate School (in press).

18 H. Yuaso and M. Ito, 'An autonomous decentralized system with application to a gait pattern generator', *Trans. Soc. Instrum. Control Eng.*, **26**(12) (1989), 180–187.