AISB Quarterly No.113

The Newsletter of the Society for the Study of Artificial Intelligence and Simulation of Behaviour

In this issue:

Features

Reviews

AISB 11

Learning lessons from biology:

From hoverfly stealth to missile camouflage

The use of camouflage and mimicry as deceptive strategies are commonplace in nature and well publicised. The subject of this article is, however, less well known: a stealth behaviour designed to camouflage motion. Motion camouflage, first suggested by Srinivasan and Davey,[1] is a technique that may be used by a predator to help conceal its motion as it approaches moving prey. The paradoxical basis of motion camouflage is that the predator should approach the prey along a route such that the predator's optic flow—as observed by the prey—resembles that of a stationary object (a fixed point) in the environment (rather than moving laterally away from the fixed point as would be the case for any non-camouflaged approach). The predator does this by ensuring it is positioned directly in between a nominal fixed point in space and the prey's current position (see Figure 1).

Srinivasan and Davey[1] observed that male hoverflies may track females in a manner consistent with motion camouflage, and that insects possess both the machinery and mechanisms to implement this strategy. We have followed on from their observations and speculations by constructing and examining an artificial neural model for this intriguing stealth behaviour. The first question posed was whether it was possible to design a (simulated) connectionist motion-camouflage control system that operates using only the information an insect can retrieve from its senses.

The answer to this question seems to be yes.[2] The only external information our control systems were supplied with was the current direction of the prey. The simulated 'fly controllers' were expected to estimate the position of the fixed point using dead reckoning based upon proprioreceptive signals (i.e. recent motor outputs were fed back as input). The control systems were tested on different prey trajectories—including those filmed of real hoverflies—and shown to adopt accurate, camouflaged approaches that predict prey motion. These results, which show that a system with workings reminiscent of a biological nervous system can perform motion camouflage with basic sensory input, are considered to support the conjecture that motion camouflage is not beyond the computational power of insects.

Having designed the control system, the next stage was to test whether these motion camouflage controllers would be able to fool humans! We did this by performing a novel psychophysical experiment that masqueraded as a computer game competition.[3] The competition was based on a 3D computer game *(Missile Defence)* of the popular first-person shoot-em-up genre: purpose-written to allow comparison of subjects' success in detecting different approach strategies. In the game, the player takes the role of the prey. The player flies along the centre of a straight tunnel (see Figure 2) whilst attempting to shoot missiles (fired at the player). The missiles represent our predators employing different strategies to approach the prey. Three different approach strategies were investigated. The first was motion camouflage. The second was a homing approach where, at each time step, the missile moved in the direction of the prey. In the third, a direct-interception ap-proach, the missile moved in a straight line to intercept the prey as quickly as possible: these missiles had access to the path of the prey in advance.

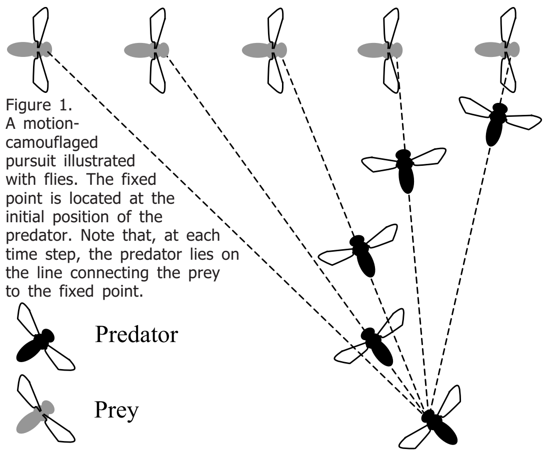


Figure 1. A motion-camouflaged pursuit illustrated with flies. The fixed point is located at the initial position of the predator. Note that, at each time step, the predator lies on the line connecting the prey to the fixed point.

The results of the experiment, conducted on 30 volunteers, showed that motion-camouflaged missiles were, in general, able to get closer to the player than missiles using the other strategies: they approached to an average distance-to-prey that was just ~60% of the missiles using the homing strategy and ~50% of those attempting direct interception. This psychophysical experiment therefore served two purposes: to further validate the design of the control system using a real task; and to provide the first evidence suggesting that humans are susceptible to motion camouflage.

The study and modelling of the hoverfly's stealth behaviour has proved useful in that it has

***Anderson/McOwan**: Queen Mary, U. London*

Continued on p. 7

References

1. M.V. Srinivasan and Davey, *Strategies for active camouflage of motion,* **Proc. R. Soc. Lond. B 259,** pp. 19-25, 1995.
2. A.J. Anderson and P.W. McOwan, *Model of a predatory stealth behaviour camouflaging motion,* **Proc. R. Soc. Lond. B 270,** pp. 489-495, 2003.
3. A.J. Anderson and P.W. McOwan, *Humans deceived by predatory stealth strategy camouflaging motion,* **Proc. R. Soc. Lond. Biology Letters,** in press.
4. A.J. Anderson, *Sensorimotor neural systems for a predatory stealth behaviour camouflaging motion,* **Ph.D. thesis, Queen Mary University of London,** submitted.

Why you (and your colleagues) should join the AISB...

The *AISB Quarterly* is published by the Society for the Study of Artificial Intelligence and Simulation of Behaviour.

ISSN 0268-4179

Editor/Managing Editor:
Sunny Bains
aisbq@aisb.org
http://www.aisb.org.uk
and click on *AISBQ*.

Advertising and Administration:
Therie Hendrey-Seabrook
aisbq@aisb.org.uk
School of Cognitive and Computing Sciences
University of Sussex
Falmer, Brighton
UK, BN1 9QH
T: (01273) 678448
F: (01273) 671320

AISB PATRON
John Barnden
University of Birmingham

If you're reading a copy of the Quarterly that you picked up at a meeting, borrowed from a friend, or downloaded from the web—and have found it worthwhile—you should think about joining us. The many benefits to joining the Society for the Study of Artificial Intelligence and the Simulation of Behaviour include:

- The AISBQ Quarterly, containing concise articles covering the latest research in the UK and beyond, book reviews, conference reports, opinion pieces, and Father Hacker.
- The AISBJ, the society's twice-yearly journal containing articles reflecting the broad spectrum of interests within the society. Members are also encouraged to submit articles to the journal.
- Our weekly e-mail bulletin, which contains the latest news, conference calls, job opportunities and information on worldwide AI events.
- Reduced registration fees at the annual AISB convention: one of the UK's biggest general AI conferences.
- Travel awards from both AISB and ECCAI
- Reduced rates to *Artificial Intelligence* (Elsevier Science) and *Artificial Intelligence Review* (Kluwer Academic Publishers).
- Access to restricted areas of AISB website.
- E-mail addresses for patrons and supporting members (yourname@aisb.org.uk).
- Assistance in the formation of local and regional AI Groups through small grants towards recruiting external speakers.
- The AISB membership fee is tax deductable.

Membership prices

There are six types of membership:

Type	UK	Europe	World
Student	£15	£18	£30
Ordinary	£30	£33	£45
Supporting	£100	£100	£100
Patron	£300	£300	£300
Institutional	£100	£100	£100
Corporate	£300	£300	£300

Patrons and *Supporting Members* are entitled to their own AISB e-mail address. *Patrons* will also be listed in the AISBQ in recognition of their contribution to the society. The extra cost of these memberships contributes to other areas of the Society e.g., travel grants for student members.

Corporate Members can advertise their corporate logo on the AISB corporate members web-page, as well as being able to place free features on the website, and free job advertisements on the AISB jobs web-page and in the AISBQ.

Any non-student SSAISB member can sign *Student Membership* forms as proof of status.

Download your membership form now from:

www.aisb.org.uk/join.html

Opinion

I recently fell out with a senior academic at my college (not in my own department, I'm happy to say). Among other things, I was annoyed that he presented a specialist presentation to an inter-disciplinary seminar. For those of us (me?) who had the wrong background, the talk quickly spiralled out of the understandable into the obscure. So much so that I gave up: I had heard him speak before with no better success but had given him the benefit of the doubt because *that* seminar really wasn't aimed at me. This time I figured I had better things to do.

Artificial Intelligence and Simulation of Behaviour are profoundly interdisciplinary subjects. So much so that I'm sometimes amazed that we can talk to each other at all. With some effort, most of us do manage: this despite the irritating loss of precision that results when handy jargon is banned. Perhaps the others doubt that this effort is worthwhile. At the risk of preaching to the converted, I'd like to make a case that it is.

In the 15 years or so that I have worked around emerging technologies, interdisciplinary research has been where the action is. Holography, my first love, was populated by photographers, physicists, artists and electrical engineers. Micro-electromechanical systems were developed by mechanical and optical engineers, materials scientists, and circuit designers. Research in neural systems has required contributions from everyone from the lab rat and the student who puts him through his paces to the computational neuroscientist and the neuromorphic engineer.

When communication between these groups breaks down, so does the productivity of research. Fast algorithms and low-power hardware are irrelevant if the researchers they're built for don't need them or can't figure out how to use them. The designer who didn't listen to what was required, or couldn't explain how the new system fit the bill, may be to blame. Or the potential user's inability to explain what they wanted might be the culpret. Either way, a communication problem can become a research problem, with talented people building toys that work well, but for no-one.

Speaking in tongues is the way to go.

Sunny Bains (Editor)
Imperial College London
http://www.sunnybains.com

Robot learning through human interactions

Research into socially-interactive robots covers many areas, and robots are designed to be so for various purposes: for instance, to promote natural human-robot interactions, or to learn about social phenomena, emotions, and development.[1] Our work concerns social interactions for learning, and we are interested in finding out what levels of interaction are required for learning to occur successfully. Is there a minimal level of interaction that is necessary? For example, if a robot could translate the actions of a human onto its own motor abilities, and therefore copy the human's actions, is this enough for the robot to learn a task that consists of these actions? Or do the interactions need to be more interactive?

Let us consider the above scenario, where the robot copies the actions of a human demonstrator as 'minimal'. (Of course, the problem of translating the actions of a human into the robot's own motor repertoire is anything but trivial, but let's assume that the robot has this ability). Where the system performs supervised learning using the robot's perceptions and actions during this copying, minimal interactions provide valuable boot-strapping that allows the machine to learn the demonstrated task.[2] This avoids many difficulties that would arise in programming the robot to perform the task: such as having to figure out what the robot actually perceives and the consequences of its actions.

However we have found that such 'minimal' interactions are not always sufficient, especially when a real physical (as opposed to simulated) robot is involved, and the learning setup is such that the robot is continuously faced with sensory-motor data that are noisy and unstructured. In such situations it is crucial that the human demonstrator take a more active and interactive role in the teaching process (see Figures 1 and 2). The human can be more active by manipulating the movements of the robot such that it is exposed to experiences that will make the learning easier, or by indicating to the robot—using explicit signals—parts of the demonstration that the human deems important. The human can be more interactive by allowing the robot's current physical state influence the demonstrations, rather than providing passive demonstrations that are independent of the robot.

Attention plays a very important role in such a learning setup. Another facet of our work therefore concerns an explicit model of an attention system that considers saliency parameters for deciding when to learn. We refer to this kind of attention as temporal (as opposed to spatial) because it involves deciding whether or not to attend to the current input based on previous experiences, rather than deciding which part of the sensory input to attend to. Having an explicit model of attention is important because it potentially allows for social interaction to contribute in an even more active way than discussed above. If the human has access to the robot's attentive state (or vice-versa, if the robot has access to the human's attentive state) then this information can be used by the robot to autonomously tune the parameters of attention. This would otherwise need to be determined by the designer (through tedious, and possibly inaccurate, trial-and-error). For example, if the robot decides to attend to an input because it is novel (one form of saliency), but the human disagrees with this response, then the robot can modify its novelty-detection threshold accordingly. Currently such attention parameters are given to our robot not through trial-and-error but through extensive systematic experimentation, which is also tedious. However, we are considering extending our work and the social interactions so that the robot can tune these parameters itself.

Thus, research into human-robot interaction is important for more than just designing believable human-like robots that will encourage humans to use them as aids, toys, companions, etc.. It is also necessary for designing systems that allow robots to learn from humans.

Figure 1. Human-robot interactions can be 'minimal' where the robot simply copies the actions of the demonstrator, who is passively performing a task such as wall-following.

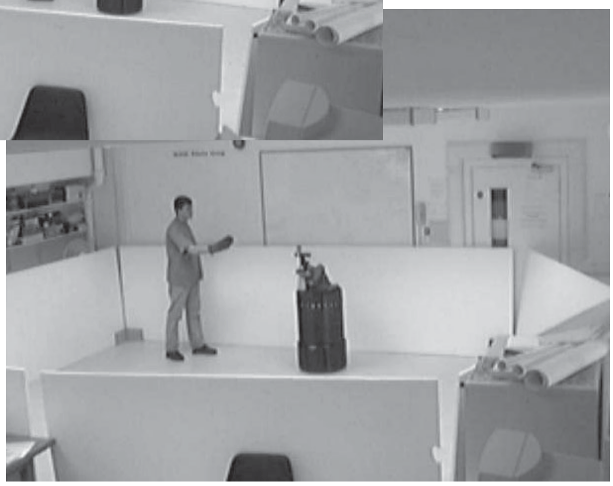


Figure 2. The human can take a more active and interactive role in the teaching process.

Yuval Marom
Institute of Perception, Action and Behaviour
School of Informatics
The University of Edinburgh
http://www.dai.ed.ac.uk/~yuvalm

References
1. T. Fong, I. Nourbakhsh, and K. Dautenhahn, *A survey of socially interactive robots,* **Robotics and Autonomous Systems 42** (3-4), Special issue on Socially Interactive Robots, pp. 143-166, 2003.
2. G.M. Hayes and J. Demiris, *A Robot Controller Using Learning by Imitation,* **Proc. 2nd Int'l Symp. on Intelligent Robotic Systems,** p. 198-204, 1994.

Are you fit for failure?

References

1. M. Garvie, A. Thompson, *Evolution of self-diagnosing hardware,* **Proc. 5th Int. Conf. on Evolvable Systems (ICES '03): From biology to hardware, LNCS 2606,** p. 238, 2003.
2. M. Garvie, A. Thompson, *Evolution of Combinational and Sequential On-Line Self-Diagnosing Hardware*, in press.
3. V. Vassilev, J. Miller, *Scalability problems of digital circuit evolution: Evolvability and efficient designs*, **2nd NASA/DoD Workshop on Evolvable Hardware,** p. 55, 2000.
4. htpp://www.cogs.susx.ac.uk/users/mmg20/contrcpu.htm
5. A. Thompson, I. Harvey, P. Husbands, *Unconstrained evolution and hard consequences*, **Towards Evolvable Hardware, LNCS 1062,** p. 136, 1996.

Self-diagnosing hardware is important: especially in mission-critical systems exposed to radiation. Built-in self test (BIST) is widely used, yet commonly requires more than 100% overhead in the form of double-redundant systems or off-line (interrupting normal operation) testing. Evolutionary methods applied to hardware have both produced circuits comparable to those designed by experts, as well as unconventional circuits in which hardware resources are used extremely efficiently. Moreover, many evolved systems in nature exhibit self-diagnostics (such as the immune system).

All this has led to the prospect that evolutionary methods could explore areas of design space that reuse hardware components so they contribute both to the circuit's main functionality and its BIST, leading to a low-overhead on-line solution. We have recently been the first to attempt the evolution of self-diagnosing hardware designs and will try to give a flavour of this work here.[1,2]

A generational genetic algorithm (GA) was used with a population of 32 individuals. The amount of conventional design knowledge used to set up the fitness evaluation function and the mapping from genotype to circuit was kept to a minimum. Evolving circuits were made up of two input logic gates and evaluated in a simple digital logic simulator where noise was introduced in order to facilitate transfer to real hardware. Hardware faults were simulated by sticking a gate output at 0 or 1, a model well established in industry.

Small circuit tasks were chosen as good starting points to establish a proof of principle that BIST functionality could be evolved for them: a one-bit full adder, a two-bit multiplier and an edge-triggered D-latch. The fitness function evaluated a number of circuit properties here listed in decreasing priority order: perform the desired task, off-line BIST, on-line BIST, minimize gate count. Self-testing behaviour was evaluated by checking if an extra output *E* went high when the task outputs were incorrect due to an induced fault. And so a process of the 'survival of the meekest' commenced.

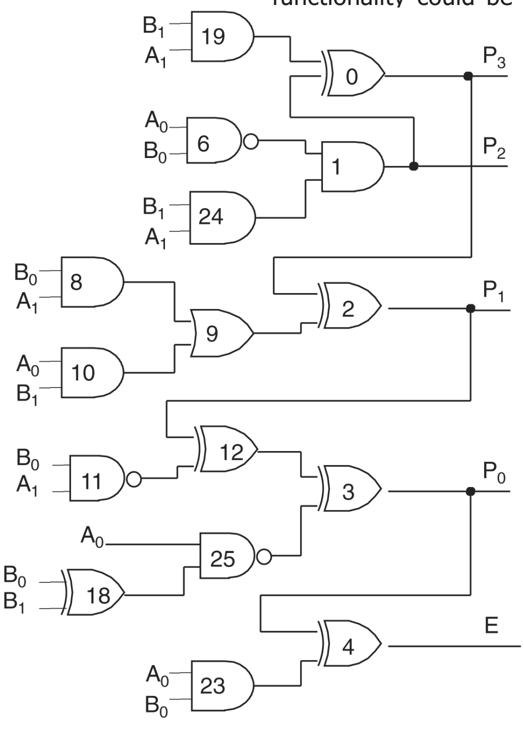


Figure 1. Evolved two-bit multiplier performs full on-line self-diagnosis with only 64% of the overhead of the conventional equivalent by reusing logic for both the main task and self-test.

From a population of random individuals, after 14100 generations of evolution, there emerged an individual performing the adder task using the minimum five gates and having 90%-fault-coverage off-line BIST using an overhead of only two extra gates. This circuit performs a hybrid of online/off-line self-diagnosis that could be implemented in a BIST system with 31% of the overhead of the conventional off-line solution. About 15000 generations later, a full (100% coverage) on-line BIST solution for the adder was found using only 50% of the overhead of the conventional on-line solution. Another run that imposed extreme noise conditions arrived at an online solution that includes a low-pass filter to iron out glitches at the output. In effect, this circuit could be clocked at twice the speed as the conventional online BIST solution.

A new run was seeded with a hand-designed multiplier using the minimum seven gates. Nearly 150000 generations later it suffered one modification while four gates were annexed for performing full off-line BIST requiring 36% of the overhead of the conventional equivalent. A multiplier with full online BIST was also evolved from a population of random genes after roughly four million generations (three weeks processing time). This circuit used 64% of the overhead of the conventional on-line solution and its unconventional structure is shown in Figure 1. An on-line self-diagnosing edge-triggered D-latch was also evolved after 3 million generations and had the same structure and overhead as the conventional solution.

These self-diagnosing circuits, evolved for the first time, are competitive with conventional ones in terms of fault coverage and gate count overhead. Evolved circuits exploit conventional design principles—such as voting and design diversity—as well as unconventional principles, such as computing checksums while cascading outputs. These principles, which allow them to reuse logic for both the main task and BIST, could prove useful if adopted by designers. Some circuits were extremely modular in structure while others were inscrutable. The reason for this is unknown but, then again, evolution moves in mysterious ways.

Previous work[3] suggests larger circuits are riper targets for evolutionary optimization, but computational power is a limiting factor when evolving them (you can easily contribute your unused CPU time to this project[4]). Our current efforts include the evolution of BIST for industry-sized modules of self-diagnosing analogue circuits, perhaps under varying operating conditions,[5] and of circuits capable of 'testing the tester' under multiple faults.

Miguel Garvie
COGS, University of Sussex
E-mail: mmg20@cogs.susx.ac.uk

Swarms and self-organized music

Improvised music differs from composed music in a very simple way - it is composed and performed simultaneously, with no chance to choose between ideas, polish up a rhythm or get a chord change right. The best known example of music with improvisation is jazz, where performers have to produce fresh melodic ideas based on a composed harmonic structure. However, experiments by avant-gardists such as John Coltrane and Ornette Coleman and—in the classical tradition—by composers such as John Cage and Cornelius Cardew, have led to forms of improvised music with very little pre-thought structure.[1] The musicians assemble and, without rehearsal or any sort of written or tacit plan, begin to develop a piece of music that is improvised at every level. Apart from important aesthetic considerations, the obvious questions are: how do humans do it and can we get a computer involved?

Answers might be found in the biologically inspired field of *self organization*. Avian flocks, insect swarms, and shoals of fish are examples of systems that develop spatio-temporal organization. In other words, global properties arise from local low-level interaction. These systems have no central control or organising plan, yet are capable of responding to a complex environment. Surprisingly, the apparently choreographed motion of flocks can be explained by assuming that relatively simple individuals interact locally with near neighbours. Raid patterns of army ants, consisting of many interconnecting trails, can contain several thousand virtually blind individuals: all acting with a common purpose, yet without central control. Once more, the explanation is to be found in the principles of self-organization.[2]

An improvising ensemble might, therefore, be subject to similar rules. In this case, the emergent structure would be musical form. The musicians respond in an expressive way to the current musical environment: trying to match current parameters such as pitch range, dynamic level, and density of events. They may also try to alter the musical direction by deliberately trying to provoke change. Such fluctuations may become amplified due to the matching behaviour of the ensemble. Hence musical structure is derived from the swarm-like interaction of the participants.

At another level, the succession of events that comprise a melody can be viewed from a swarm-like perspective. In Swarm Music, particles of a virtual swarm interact with other particles by trying to cluster together without colliding and by moving towards one or more external attractors. The organized pattern of particles in a four-dimensional physical space is interpreted as a melody by mapping positions onto a four-dimensional space of event parameters: time between events, pitch, loudness and event duration (see Figure 1).

A number of swarms are used in swarm music, each corresponding to a musical performer. Swarms themselves interact by a process known as stigmergy. Social insects may interact directly by touch, smell, etc., and also indirectly by making modifications to the environment that other individuals respond to at a later time. This indirect or stigmergetic interaction is responsible for termite task coordination and nest building. In Swarm Music, each particle in a swarm leaves behind marker cones that become attractors for other swarms. Each swarm is a separate process and the resulting system is called a 'multi-swarm' (see Figure 2).

References
1. R. Dean, **New Structures in Jazz and Improvised Music since 1960,** Open University Press, 1992
2. E. Bonabeau, M. Dorigo and G. Theraulax, **Swarm Intelligence: From Natural to Artificial Systems,** Oxford University Press, 1999.
3. T.M. Blackwell, *Swarm Music: Improvised Music with Multi-Swarms,* **Proc. AISB Symp. on AI and Creativity in Arts and Science,** pp. 4149, 2003.

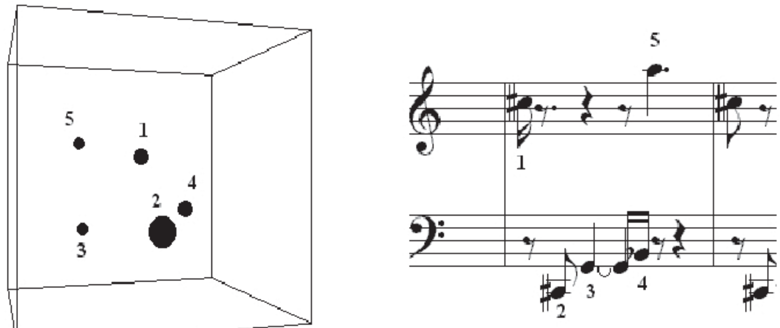


Figure 1. Interpretation (in three dimensions) of a five-particle swarm. The interpretation assumes no musical syntax and is very transparent in order to enable a direct and intuitive map from physical position to musical note.

Swarm music interacts with human performers by capturing audio and MIDI events, inversely interpreting these events, and placing corresponding cones in the physical space of each member of the multi-swarm. This means that the swarm-human interaction is identical to the swarm-swarm interaction. Some experiences of collaborators with Swarm Music have been recorded, as anecdotes, in Reference 3, which also has further information on the underlying principles and system design. Examples of autonomous and interactive Swarm Music can be found on my website.

Tim Blackwell
Department of Computer Science
Goldsmiths College, London
E-mail: tim.blackwell@ieee.org
http://www.timblackwell.com

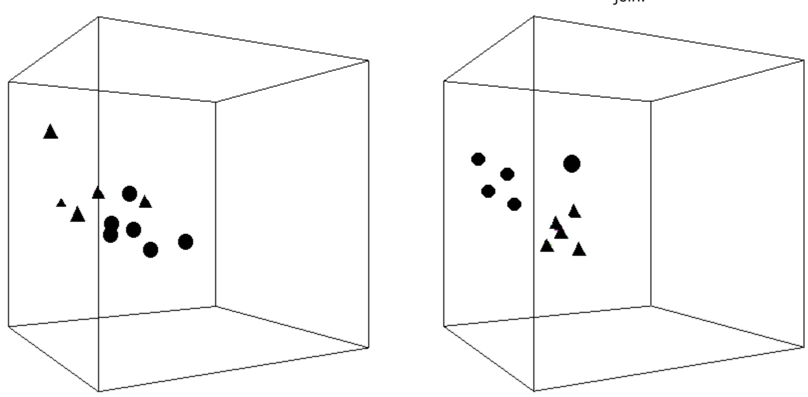


Figure 2. A two-swarm. In this snap shot, the left swarm has started to move towards the front of the cube. Attracting cones corresponding to this fluctuation are deposited in the right swarm. Both swarms may follow this new direction, or the cones deposited by the right swarm may drag the left swarm back. Musically, this corresponds to a surge in loudness and a drop in pitch, a musical direction that other members of the ensemble are invited to join.

Instructing robots

Knowledge transfer between a user and a new robot assistant is necessary to make the robot functional. In the same way, human employees need instruction before being able to fulfil their duties. In industrial robotics, robots are programmed by a small number of trained operators. However, in domestic or service robotics, potential users are many and cannot be expected to know or learn a programming language. Instead, the machines may need to be able to understand instructions issued in natural language and then convert them into the appropriate code. A joint EPSRC project between the Universities of Plymouth and Edinburgh was therefore aimed at designing a system for programming robots using spoken input, and identifying the limitations of current tools.

In the *Instruction Based Learning* (IBL) project, a mobile robot is instructed on how to travel from one place to another in a miniature town[1] (see Figure 1). On the basis of these instructions, it creates a computer program that it uses to navigate. The complete system, starting with spoken input and ending with a navigating robot, has been built and was recently exhibited in the Plymouth City Museum as part of a robotics week. In that noisy environment, speech recognition proved surprisingly robust, failing mainly on the high-pitched voice of some children. Indeed, there are still unsolved problems as outlined in the following sections.

The first step in the design of the IBL system was the recording of a corpus of 144 route instructions given by 24 subjects. They were told to speak for a human operator who would later move the robot by remote control. The transcripts of the utterances (6600 words, 330 distinct words) were used to select, out of a wide-coverage grammar, the restricted set of grammatical rules and lexicon corresponding to this domain. In principle, this user-centred approach to the design of restricted grammars enables users to employ unconstrained speech while maximizing speech-recognition performance. The results, however, showed that wide-coverage grammars do not hold for some of the forms found in spoken language: indeed, 40% of utterances were not covered. Further, there were indications that the domain lexicon was not closed and that about one new word had to be expected for every two new instructions.[2] Neither problem yet has accepted solutions, and further research is required: for instance, grammars for spoken language need to be developed, as do mechanisms for dealing with out-of-grammar words.

The corpus underwent a further functional analysis to determine which navigation actions users refer to in route instructions. A list of 14 primitive functions was established[3] including *turn in direction x after the nth landmark y* and *the goal x is located in relation y to landmark z*. Each of these functions was pre-programmed into the system, and a new procedure specification language (PSL) was created to encode rules that map natural language expressions to the appropriate function calls.[4] There are many ways to refer to a given action and about 200 rules were required. Again, the list of primitive functions was found not to be closed, with about one new function expected for every 35 new instructions. It is unclear at present how this problem should be dealt with, because a primitive is a piece of low-level robot program that the user cannot create. Here, methods of learning by example may prove useful.[5]

The combination of primitives defined in the instruction was then converted into a new piece of program code (using the scripting language Python) having the same access protocol as pre-programmed primitives. Thus, learned procedures can be re-used in later instructions to create more complex procedures (Figure 2). However, natural language references to previously taught procedures revealed a range of new problems that have only

Figure 1. Experimental set-up: a miniature remote-brained robot with a 8cm×8cm base performs vision-guided navigation in a 170cm×120cm model town. It follows a sequence of actions defined in prior instructions given by a user. Video images are sent from the robot to a PC for processing via wireless. The resulting motion commands are sent back by wireless to the robot. The user speaks to the robot through a headset microphone connected to the PC.

References
1. http://www.tech.plym.ac.uk/soc/staff/guidbugm/ibl
2. G. Bugmann, S. Lauria, T. Kyriacou, E. Klein, J. Bos and K. Coventry, *Using Verbal Instruction for Route Learning: Instruction Analysis,* **Proc. TIMR (Towards Intelligent Mobile Robots) '01,** ISSN 1361-6161, 2001.
3. S. Lauria, G. Bugmann, T. Kyriacou and E. Klein , *Mobile Robot Programming Using Natural Language,* **Robotics and Autonomous Systems 38** (3-4), pp. 171-181, 2002.

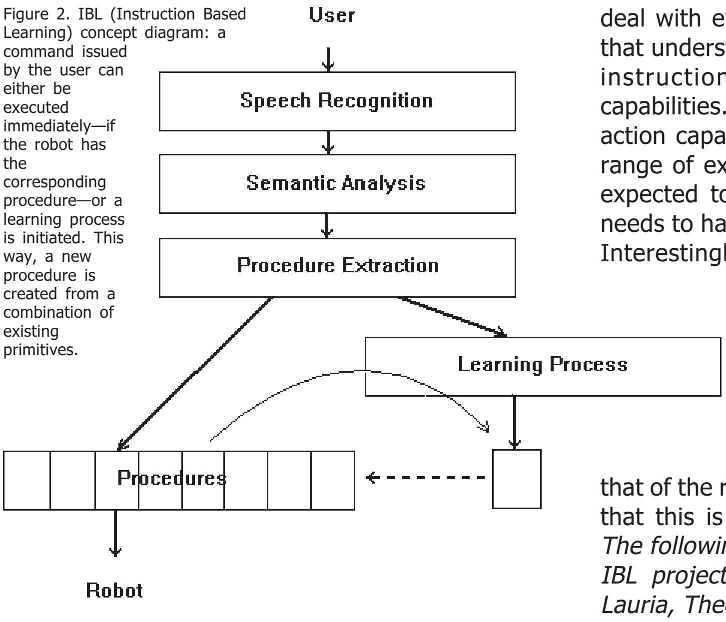


Figure 2. IBL (Instruction Based Learning) concept diagram: a command issued by the user can either be executed immediately—if the robot has the corresponding procedure—or a learning process is initiated. This way, a new procedure is created from a combination of existing primitives.

partially been solved in the IBL project.[4]

Primitives are actions referred to in human-to-human instructions. They correspond to common human execution capabilities such as finding a left turn and taking it. For robots, however, these are not simple functions and it is not straightforward to write a function that can deal with every type of left turn.[6] Thus, a robot that understands unconstrained natural language instructions needs quite advanced action capabilities. Or, conversely, a robot with limited action capabilities can only understand a limited range of expressions. As future users cannot be expected to know all the robot's capabilities, it needs to have an information mechanism built in. Interestingly, this turns out to be the same problem as teaching the user which expressions the robot can understand. Future research along these lines may also solve many of the natural language processing problems above, as the user would seek to adapt his/her language to that of the robot. Demonstrations have suggested that this is a natural tendency in users.

The following researchers also contributed to the IBL project: Ewan Klein, Johan Bos, Stanislao Lauria, Theocharis Kyriacou and Kenny Coventry.

Guido Bugmann
Robotic Intelligence Laboratory
University of Plymouth
http://www.tech.plym.ac.uk/soc/staff/guidbugm/bugmann.htm

References continued

4. S. Lauria T. Kyriacou G. Bugmann, J. Bos, and E. Klein, *Converting Natural Language Route Instructions into Robot-Executable Procedures,* **Proc. IEEE Int'l Workshop on Robot and Human Interactive Communication (Roman '02),** Berlin, Germany, pp. 223-228, 2002.
5. S. Schaal, A.J. Ijspeert and A. Billard, *Computational Approaches to Motor Learning by Imitation,* **Philosophical Transactions: Biological Sciences 358 (1431),** pp.537-547, 2003.
6. T. Kyriacou, G. Bugmann, S. Lauria, *Vision-Based Urban Navigation Procedures for Verbally Instructed Robots,* **Proc. IEEE/RSJ Int'l Conf. on Intelligent Robots and Systems (IROS '02),** Lausanne, Switzerland, pp.1326-1331, October 2002.

Lessons from biology: Cont. from cover

demonstrated the simplicity of the input information required to accurately calculate camouflaged approaches. In addition, it has provided evidence to suggest that motion camouflage has real-world applications. The most likely artificial applications of motion camouflage are military: for instance as an automatic control strategy for missiles or aircraft. Possible extensions of motion camouflage to counter radar and thermal imaging are also possible.[4] The other obvious application for motion camouflage, as demonstrated by the experiment discussed above, would be in the AI of predatory agents in computer games.

Andrew J. Anderson[a] and Peter W. McOwan[b]
Department of Computer Science
Queen Mary, University of London
a. E-mail: aja@dcs.qmul.ac.uk
http://www.dcs.qmul.ac.uk/~aja
b. E-mail: pmco@dcs.qmul.ac.uk
http://www.dcs.qmul.ac.uk/~pmco

References on cover

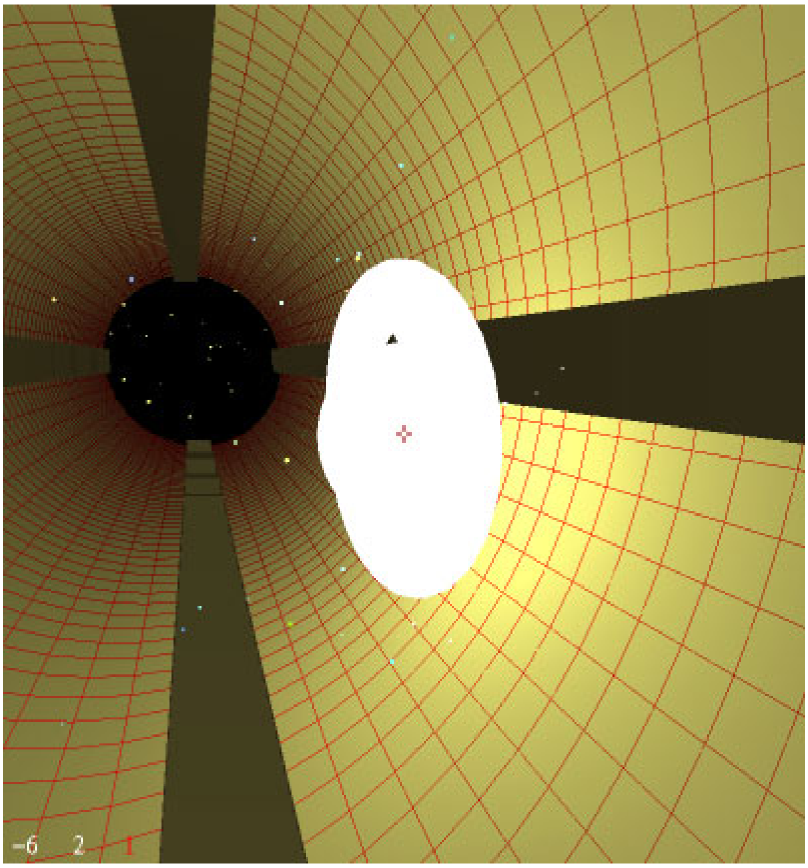


Figure 2. *Missile Defence* screenshot, an explosion as the player hits a target. (This screenshot and others are currently displayed at http://www.dcs.qmul.ac.uk/~aja).

The evolution of optimism:

An agent-based model of adaptive bias

Dealing with uncertainty is a common problem for agent systems. The ability to reason with uncertain information is an indispensable requirement for modelling intelligent behaviour in a complex and dynamic environment. This is why uncertain reasoning has become a major research topic in AI, with many important applications.

Psychologists have shown that human judgement under uncertainty involves consistent departures from normative rationality. In particular, people show ‘motivational biases’ in judgements of probability, over-estimating the probability of events with a positive return to the self and under-estimating the negative.[1,2]

From the standpoint of rational choice theory, these biases are clearly maladaptive. Some psychologists, however, have argued that they are adaptive.[3] We have attempted to adjudicate between these two possibilities by constructing a multi-agent based computer model in which a variety of agents with different decision rules are allowed to compete in various environments.

Methods

We designed a multi-agent-based simulation using the program NetLogo.[4] The agents in our model face ‘opportunities’, each of which has a probability of success (p), a benefit for success (b) and a cost of failure (c). Different noise levels influence the agents’ knowledge of c, b and p.

For every opportunity faced, each agent must decide whether or not to 'play'. This decision is made according to the agent's decision rule. There are three types of agent:

1. The *rational* agent uses the principle of expected utility. That is, it only plays when:
$p * b > (1-p) * c$.
2. The way the *optimistic* agent works is inspired the empirical data from situations where humans' judgement is biased. People's probability weightings follow an inverse S-shaped curve.[5] We modelled this by giving our optimistic agents a biased estimate of p. This agent plays when: $(p * (b/c)) * b > ((1-p) * (b/c)) * c$.
3. The *emotional* agent is based on the observation that people tend to play when the benefits are high: independent of the probability of success. The lottery is a case in point. But it also works the other way around, so this agent always plays if $b/c > 2$ and never plays if $b/c < 0.5$. Only when the difference between cost and benefit is small do people seem to attend to the probability of success.[6] We modelled this by stipulating that when $2 > b/c > 0.5$, the emotional agent’s chance of playing is proportional to its estimate of p (random $1 < p$).

Each agent’s chance of success is determined by the probability associated with the opportunity in question. If an agent plays and succeeds, its energy level is increased by the benefit for success associated with the opportunity. If it plays and fails, its energy level is decreased by the cost of failure. If an agent does not play, its energy level remains the same.

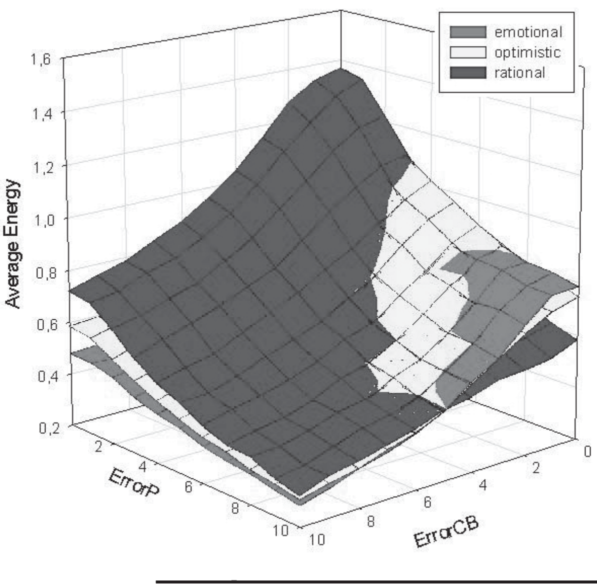


Figure 1. 3D-Graph showing the average energy for each of the three kinds of agents under different error conditions.

Results and conclusion

We let the program run 10 times for every possible combination of the errors of p, c, and b. Figure 1 gives an overall view of our results. Not surprisingly, the rational agents do better than all other agents under most conditions. More interesting is the fact that there are conditions under which the rational agent is outperformed. The biased agents do better when the error for b and c is low, but the error for p is high.

This is arguably the situation that people mostly encounter in the real world. It is plausible to think that people can estimate costs and benefits of an opportunity quite accurately, by observing other people faced with similar opportunities and by memories of past experiences. However, the chances of success of any specific opportunity depend on the interaction with other human beings and many other imponderable factors. Hence our ability to estimate probability is much poorer than our ability to estimate the cost of failure or the benefit of success.

From the standpoint of classical decision theory, motivational biases are clearly irrational. We have found that—under certain environmental conditions—biased agents that behave in ways similar to humans outperform classically rational agents acting purely to maximize expected utility. Our findings, therefore, support the view that motivational biases are adaptive.

Our paper and model can be found at:
http://www.dylan.org.uk/OptimismAISB.html

Dylan Evans[a], **Annerieke Heuvelink**[b], **and Daniel Nettle**[c]
a. University of Bath, UK
E-mail: D.Evans@bath.ac.uk
http://www.dylan.org.uk
b. University of Utrecht, The Netherlands
E-mail: A.Heuvelink@phil.uu.nl
c. The Open University, UK
E-mail: D.Nettle@open.ac.uk.

References

1. D.T. Miller and M. Ross, *Self-serving biases in attribution of causality: Fact or fiction?,* **Psychological Bulletin 82,** pp. 213-225, 1975.
2. M. Zuckerman, *Attribution of success and failure revisited: The motivational bias is alive and well in attribution theory,* **Journal of Personality 47,** pp. 245-287, 1979.
3. S.E. Taylor and J.D. Brown, *Illusion and well-being: a social psychological perspective on mental health,* **Psychological Bulletin 103,** pp. 193-201, 1988.
4. NetLogo: http://ccl.northwestern.edu/netlogo/
5. D. Kahneman and A. Tversky, *Choices, values and frames,* **American Psychologist 39,** pp. 341-350, 1984.
6. Y. Rottenstreich and K. Hsee, *Money, kisses and electric shocks: On the affective psychology of risk,* **Psychological Science 12,** pp. 185-190, 2001.

Review

Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison

Edited by David Sankoff and Joseph Kruskal

Thanks to advances in molecular biology—including molecular cloning, DNA sequencing, and the human genome project—interest in sequence comparison has recently increased. However, these techniques can be used more widely: to sequence agent plans or robotic behaviours, for instance. The book *Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparison* was first published in 1983 and was reprinted in paperback in 1999. While other algorithms for sequence comparison have been published in the literature, this book is the definitive source. It describes techniques for comparing sequences by measuring the Levenshtein or edit distance (defined as the minimum number of operations required to change one sequence into another). With discrete genetic sequences, these can be used to measure deviations or mutations. Also, continuous time-dependent signals can be compressed or expanded (time warped) and then sampled: this process has been used in speech recognition.

How can the artificial-intelligence community use sequence comparison? Consider the problem of merging partial plans, where each plan is a sequence of discrete actions. If each action has a function or functions associated with it that describe the costs of deleting, delaying, or modifying that action, then the Levenshtein distance, with each operation weighted by the appropriate cost function(s), could be used to develop the best merged plan. A related problem is recursive learning; applying 'rewards' or 'penalties' to a sequence of actions that have led to a successful or unsuccessful outcome. With traditional techniques it can be difficult to assign rewards based on order, particularly if the ordered pair is not contiguous. Using the edit distance, on the other hand, allows wide latitude in how reward is applied and how state transitions are learned.

Time Warps... is written at the graduate level with articles covering the fields of computer science, genetics, linguistics, and speech recognition. While some edited books can be uneven in the depth and breadth of coverage from chapter to chapter, this book has a flow that is usually only achieved by a single author. Although the contributors are diverse, both geographically and academically, the reader gets the impression that this book is actually a collaboration, not a collection, of essays. Though I have not tested the algorithms, on inspection they appear complete enough that computer programs could be readily developed.

The introduction to the new edition, Edit Distance and Dialect Proximity, could easily serve as an additional chapter. It uses Levenshtein distance as a measure of phonetic variation of Dutch dialect. Overlaying the phonetic distance with the geographic map of the Netherlands shows a continuum of dialect that non-numeric methods have been unable to demonstrate.

Part One focuses on genetic application, presenting algorithms for solving the problem of finding a section of a long sequence, `with best possible agreement', to a shorter sequence, where best possible agreement can mean shortest overall edit distance or, in the case of repeating sequences, the shortest 'local' edit distance.

Time warping of continuous functions of time, the process of stretching or compressing the time axis, is discussed in Part Two. Extensively used in speech processing, this transform is generally done in the continuous domain, and is often performed during sampling to create a discrete function. Time warping can be done globally, where the entire function is stretched or compressed, or locally, where the time scale factor varies from segment to segment. Another form of time warping involves breaking the continuous function into segments of varying lengths, coding each segment, and compiling the codes into a sequence.

Part Three provides variations of the dynamic programming algorithm to address similar but distinct problems: matching a portion of a sequence to a known template; finding similar sub-sequences in different strings; simultaneous comparison of multiple sequences; auto-comparison; comparison of two trees, directed networks, or continuous functions with time warping; and comparison of two sequences under constraints. Other variations included are sequence comparison where adjacent elements may be transposed, and where more generalized transpositions and substitutions are permitted.

Computational complexity is addressed in Part Four (along with Chapter 7 in Part Three). While two of the three chapters are necessarily theoretical, Chapter 14 presents an edit distance algorithm that is faster for long sequences and a cost function to calculate when it is worth using.

Part Five discusses methods of determining whether the similarity between two sequences is meaningful or random. Upper and lower expected similarity bounds are computed for random sequences and reworked in terms of sequence distances. These can be used in a standard 'null partial' plans in multi-agent systems, adaptive control, and stock market analysis. Nor does it discuss how advances in neural-network programming have reduced the computational load of—thereby increasing interest in—sequence analysis algorithms.

Mike Roeckel
Applied Research Laboratory
The Pennsylvania State University
E-mail: mwr3@psu.edu

Publisher:
CSLI Publications
Stanford University
http://www-csli.stanford.edu
Pages:
382
Paperback:
ISBN 1-57586-217-4
$23.95

Conference Report

Imitation in Animals and Artifacts

Part of AISB'03, Aberystwyth, Wales, 2003

Reference
1. More details about the symposium—including the programme, links to publications related to imitation, and an MIT Press book on the topic—can be found at: http://homepages.feis.herts.ac.uk/~nehaniv/aisb03.html

In the life sciences, imitation has been studied for many years for the insight it gives into the brain mechanisms underlying visual perception of biological motion, as well as memory and motor control. It has also attracted the attention of robotics researchers because of its promise of easily-programmable robots that require a mere demonstration to learn how to perform a given task. It was in this climate that the *Second International Symposium on Imitation in Animals and Artifacts*[1] was held as part of the AISB convention this year. It was the longest of the symposia, spanning all five days of the convention, and attracted a large number of participants from as far as Australia and North America.

Following the successful format of the first symposium (at AISB'99), it inter-mixed invited and refereed papers from a variety of disciplines, including contributions from animal behaviour, psychology, philosophy, brain imaging, computer science, and robotics. This was mainly due to the efforts of the programme chairs, Kerstin Dautenhahn and Chrystopher Nehaniv of the University of Hertfordshire, who should be congratulated for rigorously enforcing the interdisciplinary focus of the meeting through their selection of invited speakers and contributed papers. Rather than grouping the papers into themes, Dautenhahn and Nehaniv chose to interlace talks from the various disciplines in all five days, combatting the trend usually observed in conferences of participants attending only the sessions directly relevant to their work.

The participants responded well to this arrangement. It was clear that the presenters (and particularly the invited speakers) had understood the need to explain their work to an interdisciplinary audience, and did so eloquently. Combined talks, where researchers from different disciplines took turns in presenting material, also made their debut in this symposium: with excellent results. For example, Jacqueline Nadel (developmental psychology) and Arnaud Revel (robotics) examined how information from infant psychology can be used to advance developmental approaches to robotic imitation. Sarah Woods and Kerstin Dautenhahn combined efforts pursuing the interplay between bullying behaviour, empathy, and imitation.

Another positive trend was the shift from debating definitions of imitation to the detailed examination of the mechanisms underlying imitation capabilities and their pathologies. Although the occasional definition debate did crop up (as it regularly did in the first symposium), there was a distinct emphasis on the 'how' of imitation. From Robert Mitchell's models of mirror self recognition—with an emphasis on kinaesthetic-visual matching—to John Laird's algorithms for implementing learning by observation, there was a distinct 'mechanistic' flavour to the symposium. This was complemented by presentations on the pathologies of these mechanisms: work with autistic children by Peter Hobson, Justin Williams, Jacqueline Nadel, Kerstin Dautenhahn, Jessica Meyer and Dominic Massaro.

Invited speakers included John Laird, Andrew Whiten, Aude Billard, Ludwig Huber, Robert Mitchell, Mark Norman, and myself, all supported by an EPSRC grant to the symposium chairs. With Kerstin Dautenhahn, Chrystopher Nehaniv, Peter Hobson and Irene Pepperberg, the invited speakers formed a panel for a discussion at the end of the third day. The idea was for each to pose research challenges to those from other disciplines: psychologists, for example, suggested that it would be particularly helpful to researchers interested in autism if roboticists paid more attention to the development of the imitation mechanisms observed in humans.

We left the conference and Aberystwyth with lots of new ideas, and an agreement to hold this meeting more frequently: I, for one, am looking forward to the next one!

Yiannis Demiris
Imperial College London
http://www.iis.ee.ic.ac.uk/~y.demiris/

Treasurer's report

Following the trend set in previous years, AISB has continued to make a profit in the year 2002, despite the significant cost of producing and distributing the new AISB Journal. This major cost item has mainly been covered by the success of AISB'02 held at Imperial College, which made a surplus of over £6000: thanks to the convention organisor, Dr Jim Cunningham.

Over the years, AISB has now built up a healthy reserve of £25,563. This will allow AISB to take on more adventurous projects, where appropriate, in the coming years.

Paul Chung
Treasurer

INCOME AND EXPENDITURE ACCOUNT	**2002**	**2001**
Turnover	28,052	26,996
Direct costs	(9,569)	(7,730)
EIE[1] before overheads	**18,483**	**19,266**
Administrative expenses	(15,794)	(12,251)
EIE[1] before taxation	**2,689**	**7,015**
Taxation	(6)	(30)
EIE[1] for the year	**2,683**	**6,985**
Retained profit brought forward	8,987	2,002
Retained profit carried forward	**11,670**	**8,987**

BALANCE SHEET	**2002**	**2001**
Debtors	7,849	10
Cash in bank	23,660	27,296
Current assets	**31,509**	**27,306**
Creditors: amounts falling due within one year	(5,946)	(4,426)
Total assets less current liabilities	**25,563**	**22,880**
Reserves		
Other reserves	13,893	13,893
Income and expenditure reserve	11,670	8,987
Total reserves	**25,563**	**22,880**

INCOME AND EXPENDITURE IN DETAIL	**2002**	**2001**
AISB Convention this year: income	15,612	13,858
AISB Convention this year: costs	(9,569)	(7,730)
AISB Convention this year: net	**6,043**	**6,128**
ECAI'98 Conference	-	513
AISB Convention 2001 (Correction)	(672)	-
Membership fees	12,200	11,121
Inserts in AISBQ	309	1,200
Workshop Proceedings	347	-
Gross interest received	256	304
EIE[1] before overheads and taxation	**18,483**	**19,266**

OVERHEAD EXPENSES	**2002**	**2001**
Office costs	4,930	5,373
Newsletter and Journal: production	6,969	2,526
Newsletter and Journal: distribution	1,118	602
Postage costs	471	495
Committee expenses	209	780
Computer costs	-	111
Sundry expenses	14	330
Travel awards	200	300
ECCAI membership fee	719	524
Accountancy fee	1,058	1,011
Bank charges	106	199
Total overhead expenses	**(15,794)**	**(12,251)**
EIE[1] before taxation	**2,689**	**7,015**
Taxation	(6)	(30)
NET PROFIT	**2,683**	**6,985**

Reference
1. Excess of income over expenditure.

Correction

In Aaron Sloman's article in the last issue, *How to build a human-like mind,* some text was inadvertently lost in the layout process. In the paragraph starting, *"In particular, if meta-management...,"* and after the words, *"leading robot philosophers with this sort of architecture to discover the problem(s) of..."* the following text should be inserted.

"...'qualia'. Some of them would become dualist philosophers.

It's a huge project: is it doable? I don't know. One way to make progress is to set up a long term target, and then identify a succession of increasingly difficult steps leading towards that target.

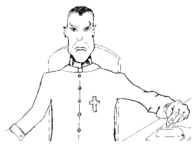
Such a target might be the design and implementation of a robot with a large subset of the abilities of a typical four or five year old child.

We can then attempt to achieve increasingly large subsets of those abilities. This contrasts with..."

The full text is available at: http://www.cs.bham.ac.uk/research/cogaff/sloman-aisbq-03/

Apologies both to readers and to Professor Sloman.

Sunny Bains
Editor



Father Hacker's Guide

for the Young AI Researcher

Cognitive Divinity Programme
Institute of Applied Epistemology

About the Society
The Society for the Study of Artificial Intelligence and Simulation of Behaviour (AISB) is the UK's largest and foremost Artificial Intelligence society. It is also one of the oldest establised such organisations in the world.

The Society has an international membership of hundreds drawn from academia and industry. Membership of AISB is open to anyone with interests in artificial intelligence and cognitive and computing sciences.

AISB membership includes the following benefits:

- Quarterly newsletter
- Biannual Journal
- Travel grants to attend conferences
- Discounted rates at AISB events and conventions
- Discounted rates on various publications

You can join the AISB online via:
http://www.aisb.org.uk

AI research is founded on computer programs. Your research reputation ultimately depends on the quality of your software. So it is vital to know...

7. How to Write Computing Programs

1. The short answer is don't: that's what graduate students are for. Your time is better spent on setting strategic research directions. However, since you have to supervise the students, you must at least be able to bluff your way through the art of programming, so read on.
2. Software engineering defines a whole paraphernalia of techniques for developing computer programs: requirements capture, specification, validation, verification, formal methods, modularisation, top-down programming, testing. For people of your ability and that of your research group, it is insulting to suggest that you need such support in the simple matter of computer programming. You can safely ignore it all.

Does your implementation lack an underpinning mathematical theory?

Let Hacker Enterprises' CONS™ (Constructs Obtuse but Nonsensical Squiggles) pepper your papers with proofs.

3. AI pioneered the technique of *exploratory programming*: the evolution of innovative systems via the run, debug, edit cycle. Critics have maintained that this leads to brittle, unreliable and flabby programs. But human intelligence has evolved in this same exploratory way with similar results, so no *neat* program could possibly provide a psychologically valid model. Exploratory programming is *essential* for accurate emulation of natural cognition.
4. Computer programs are classically evaluated by systematic, thorough-going and exhaustive testing. But this is only necessary for those who lack confidence in the key ideas underpinning their implementation. For researchers of our calibre, such excessive testing is an extravagant luxury.
5. In this age of the Internet, computer programmers frequently make their programs available as freeware over the web, allowing potential users to assess their functionality and usability. Unfortunately, this creates an opportunity for misunderstandings about the more imaginative claims you have made for your system. To protect against such misunderstandings, you should ensure that your system is only guaranteed to work with an unobtainable version of its implementation language, e.g. if the version numbers jumped from 3.12 to 4.0 then insist on version 3.13.

Protect your system from ill-informed criticism...

...with Hacker Enterprises' GREMLINS™ (Guard against Reviewers with an Enveloping Mask that Limits and Impedes Negative Scrutiny). It will output convincing 'software rot' error messages to anyone subjecting it to unfair tests.

6. Funding agencies and other reviewers are increasingly assessing the value of systems by the size of the user community you have built up. Clearly, this creates difficulties for researchers, such as ourselves, who are ahead of their time. Hacker Enterprises have addressed this problem with CLUB™ (Collaboration of Lots of User Back-scratching) our self-help users club. Subscribers to CLUB™ will all *claim* to use each other's systems, thus increasing the size of everyone's user community without the unwanted burden of providing a maintenance service.

Next AISBQ deadlines:
8 August and
14 November 2003

The AISBQ reaches hundreds of Artificial Intelligence and Cognitive Science researchers in the UK, Europe and beyond.

Advertising and other information is available via:
http://www.aisb.org.uk/qhome.html

This page includes full guidelines for the submission of book reviews, conference reviews, and technical articles. Books available for AISB members to review are also listed.