Issues in Evolutionary Robotics

I. Harvey, P. Husbands, D. Cliff

CSRP 219, July 1992

Cognitive Science Research Paper

Serial No. CSRP 219

The University of Sussex
School of Cognitive and Computing Sciences
Falmer
BRIGHTON BN1 9QH
England, U.K.

A version of this paper appears in:
Proceedings of SAB92,
the Second International Conference on Simulation of Adaptive Beha,dcands,

Issues in Evolutionary Robotics

Inman Harvey[1]

tions with the environment, and between separate parts of the robot itself (8, 22). Designing appropriate cognitive architectures is a task with inherently explosive complexity. Complexity is likely to scale much faster than the number of layers or modules within the architecture — it can scale with the number of possible interactions between modules.

To design cognitive architectures for robots with emergent behaviours hence requires either (a) a computationally intractable planning problem (10) or (b) a creative act on the part of the designer — which is to be greatly admired, though impossible to formalise. In both cases it seems likely that the limits of feasibility for real robots doing useful things are currently being reached.

3 Let's evolve robots instead

If, however, some objective fitness function can be derived for any given architecture, there is the possibility of automatic evolution of the architecture without explicit design. Natural evolution is the existence proof for the viability of this approach, given appropriate resources. Genetic Algorithms (GAs) (12) use ideas borrowed from evolution in order to solve problems in highly complex search spaces, and it is here suggested that GAs, suitably extended in their application, are a means of evading the problems mentioned in the previous section.

The artificial evolution approach will maintain a population of viable genotypes (chromosomes), coding for cognitive architectures, which will be inter-bred and mutated according to a selection pressure. This pressure will be controlled by a task-oriented evaluation function: the better the robot performs its task the more evolutionarily favoured is its cognitive architecture. Rather than attempting to hand-design a system to perform a particular task

In the long term, as the robots become more sophisticated and their worlds more dynamic, will the simulation run out of steam? The simulation of a medium resolution visual system with, for instance, motion detection preprocessing is painfully slow on today's hardware. Techniques to test many generations of control systems in real worlds will have to be developed. We are currently pursuing the development of one such technique: see (11) for further details.

6 What should we evolve?

So far we have not addressed the question of what exactly it is that is being evolved. There are at least three useful ways to implement the control system of an autonomous robot:

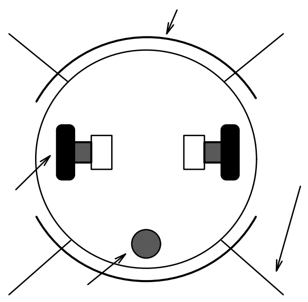
- An explicit control program, in some high level language;
- A mathematical expression mapping inputs to outputs, e.g. a polynomial transfer function;
- A blue-print for a *processing structure*, a network of simple processing elements.

6.1 High Level Programs

In (9), following a suggestion by Langton, Brooks proposes using an extension of Koza's genetic programming techniques (18) as the method for evolving a physical or simulated robot.

One potential problem with evolving a programming language is that, if it supports partial recursion, programs to be evaluated may never halt, unless some arbitrary 'time-out' is imposed. Brooks' Behaviour Language (7) does not use partial recursion, and hence can be evolved without this problem. Subject to the qualification that Genetic Programming should have genotype length changes restricted to small steps his approach at first sight seems reasonable, but we have two broad objections.

The first is that any such programming approach treats the 'brain' as a computational system, producing a set of motor outputs for any gi[illegible])Tj31.43980Td(in-)Tj-232.3212Td(puts.)Tj25.919



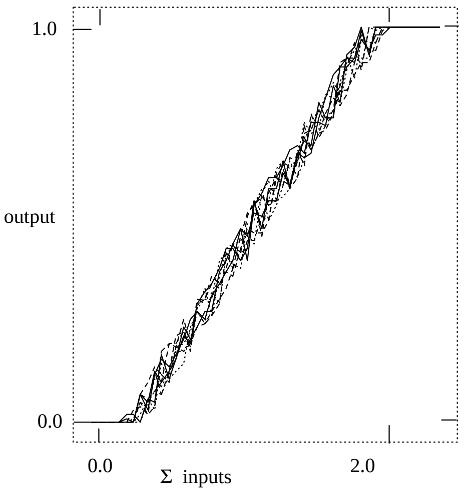


Figure 2: Noisy neuron transfer function.

specialised graphics pipeline processors) is readily available, it is envisaged that physical

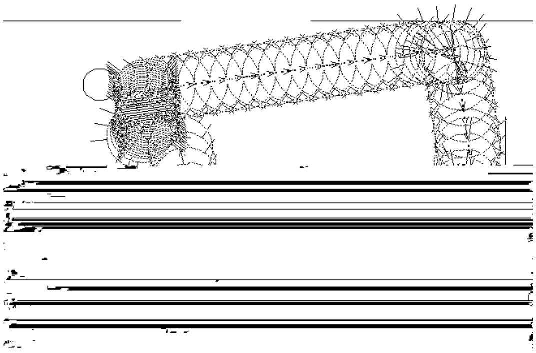


Figure 6: Motion of a robot evolved to maximise the area of the bounding polygon of its path over a limited time period.

and average much closer together, providing a far more robust solution.

Figure 8 shows a network evolved in this second experiment. It is fairly complex with many feedback loops, but it is interpretable in terms of generated behaviours. If it reminds you of a bowl of spaghetti without the bolognese sauce and chianti, this is probably partly due to the fact that there is no term in the evaluation functions that penalises unnecessary links. However, initial populations are started with individuals having (randomly) one or zero internal nodes; the number can only grow gradually if that promotes greater fitness. We expect that more concise networks will result if we introduce a cost for link creation in the evaluation function, and allow for the possibility of non-unity time delays and/or weights on connections.

Theseeigh2

[5] R.D. Beer. A dynamical systems perspective on autonomous agents. Technical Report CES-92-11, Case Western Reserve University, Cleveland, Ohio, 1992.

[6] R. A. Brooks. Achieving artificial intelligence through building robots. A.I. Memo 899, M.I.T. A.I. Lab, May 1986.

[7] R. A. Brooks. The behavior language. A.I. Memo 1227, MIT AI Lab, 1990.

[8] R.A. Brooks. Intelligence without representation. *Artificial Intelligence*, 47:139–159, 1991.

[9] Rodney A. Brooks. Artificial life and real robots. In *Proceedings of the First European Conference on Artificial Life*. MIT Press/Bradford Books, Cambridge, MA, 1992.

[10] David Chapman. Planning for conjuctive goals. *Artificial Intelligence*, 32(3):333–377, 1987.

[11] D. T. Cliff, P. Husbands, and I. Harvey. Evolving visually guided robots, 1992.

[12] David E. Goldberg. *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley, Reading, Massachusetts, USA, 1989.

[13] Steven A. Harp, T. Samad, and A. Guha. Towards the genetic synthesis of neural networks. In J. D. Schaffer, editor, *Proceedings of the Third International Conference on Genetic Algorithms*, San Mateo, California, 1989. Morgan Kaufmann.

[14] Inman Harvey. Species adaptation genetic algorithms: The basis for a continuing SAGA. In F. J. Varela and P. Bourgine, editors, *Toward a Practice of Autonomous Systems: Proceedings of the First European Conference on Artificial Life*, pages 346–354. MIT Press/Bradford Books, Cambridge, MA, 1992.

[15] P. Husbands and I. Harvey. Evolution versus design: Controlling autonomous robots. In *Integrating Perception, Planning and Action, Proceedings of 3rd Annual Conference on Artificial Intelligence, Simulation and Planning*, pages 139–146. IEEE Press, 1992.

[16] Timothy Johnson and Philip Husbands. System identification using genetic algorithms. In H.-P. Schwefel and R. Männer, editors, *Parallel Problem Solving from Nature*. Springer-Verlag, Lecture Notes in Computer Science Vol. 496, 1991.

[17] Michel Kerszberg and Aviv Bergman. The evolution of data processing abilities in competing automata. In Rodney M. J. Cotterill, editor, *Computer Simulation in Brain Science*, pages 249–259. Cambridge University Press, 1988.

[18] John R. Koza. Genetic programming: A paradigm for genetically breeding populations of computer programs to solve problems. Technical Report STAN-CS-90-1314, Department of Computer Science, Stanford University, 1990.

[19] H.R. Maturana and F.J. Varela. *The Tree of Knowledge: The Biological Roots of Human Understanding*. Shambhala Press, Boston, 1987.

[20] J. L. McClelland and D. E. Rumelhart, editors. *Explorations in Parallel Distributed Processing*. MIT Press/Bradford Books, Cambridge Massachusetts, 1986.

[21] Geoffrey F. Miller, P. M. Todd, and S. U. Hegde. Designing neural networks using genetic algorithms. In J. D. Schaffer, editor, *Proceedings of the Third International Conference on Genetic Algorithms*, San Mateo, California, 1989. Morgan Kaufmann.

[22] H.P. Moravec. The Stanford Cart and the CMU Rover. In *Proc. of IEEE*, volume 71,