A Role for Immunology in "Next Generation" Robot Controllers

Emma Hart, Peter Ross, Andrew Webb, and Alistair Lawson

Napier University, Scotland, UK
{e.hart,p.ross,a.webb,al.lawson}@napier.ac.uk

Abstract. Much of current robot research is about learning tasks in which the task to be achieved is pre-specified, a suitable technology for the task is chosen and the learning process is then experimentally investigated. A more interesting research question is how can robot be provided with an architecture that would enable it to developmentally 'grow-up' and accomplish complex tasks by building on basic built-in capabilities. Previous work by the authors defined the requirements of a robot architecture that would enable this to happen – in this paper, we describe how some components of such an architecture can be achieved using an immune network model, and present preliminary results that show the plausibility of the suggested approach.

1 Introduction

A great deal of current research work in mobile robotics and autonomous systems is still focused on getting a robot to learn to do some task such as pushing an object to a known location or running as fast as possible over rough ground. The learning process may be supervised, unsupervised or a process of occasional reinforcement, but the whole aim in such work is to get the robot to achieve the task that was pre-defined by the researcher.

As a step towards achieving truly autonomous robots that can function productively for long periods in unpredictable environments, it is important to investigate how one might design robots that are capable of 'growing up' through experience. By this, we mean that the robot starts with only some basic skills such as an ability to move about and an ability to sense and react to the world , but in the course of time it develops genuinely new skills that were not entirely engineered into it at the start. In particular it should be capable of building some kind of hierarchy of skills, such that for each new skill s_{new} there is one or more sets of skills $S_1, S_2, \cdots S_n$ such that s_{new} is significantly more easily acquired if the robot has acquired all the members of some S_i than if it lacks at least one member of each of those sets. To achieve this requires a fundamental shift in thinking when designing robotic architectures compared to the type of systems prevalent in the literature today.

Previous work by the authors [1] attempted to lay out a research agenda by which this question could be answered and identified six essential ingredients of an architecture that can realise growing-up robots. These are: sensors, memory, data-abstraction, planning, motivation, and finally a developmental schedule.

J. Timmis et al. (Eds.): ICARIS 2003, LNCS 2787, pp. 46–56, 2003.

[1] provides an overview of existing developmental architectures in relation to the above features. In this paper, we argue that an immune-network model can form the central component of a new architecture, which in particular provides a convenient method for handling the first four requirements. The immune network model was first proposed by Jerne in [7], and suggested that antibodies not only recognise foreign antigens, but also are connected together in a large-scale network formed by chains of stimulation and suppression between communicating antibodies. Although still controversial in immunological circles, the model has been successfully adopted by many AIS practitioners, producing diverse applications from data-mining systems [16] to simple robot-control architectures [4,10,14].

In the next sections, we describe the proposed architecture in detail and provide results of some early experimentation. Although this in no way represents the complete architecture and is tested only in simulation, it does at least point to the plausibility of the model.

2 Previous Work

AIS ideas have already appeared in robotics research. Lee [9] proposed an AIS for realisation of cooperative strategies and group behaviour in collections of mobile robots, and Singh and Thayer [13,15] proposed another architecture for coordination and control of large scale distributed robot teams based on concepts from the immune system. Of more relevance to this research is the work of Ishiguro and Watanabe who introduce an immune-network for behaviour-arbitration in [4,17], for gait-control in walking robots [5] and also the work of [10] who also consider an immune network for decentralised autonomous navigation in a robot. In some senses, this work suffers from the same problems as other robotic approaches in that it results in a control module that is essentially static, i.e. successfully implements certain fixed behaviours, but would not permit a robot to 'grow-up' in the developmental sense outlined in the introduction. However, the overall approach contains many elements that can be incorporated into our proposed system and hence is briefly outlined here.

In [4,17], antibodies are formed into a network that successfully arbitrates between simple behaviours on a real robot; initially they handcrafted antibodies, in later work they evolved them. An antibody consists of a paratope defining a desirable condition and related motor-action, and an idiotope which identifies other antibodies to which the idiotope is connected. Connection between the idiotope of one antibody x and the paratope of an antibody y stimulates the antibody y, and links between antibodies in the network can either be evolved by a genetic algorithm [17] or formed via an on-line adaptation mechanism which provides reinforcement signal to links, [5]. The architecture which we propose must also handle behaviour arbitration, however we wish to construct it in such a way that its links also express *sequences* of actions, and thus paths in the network represent both a past history of robot actions (i.e. an episodic memory) and also provide information useful for planning.

A related line of research to AIS is that of the application of *classifier systems* to robot-control. Rules in a classifier system consist of conditions which are matched against the current state of the environment, and associated actions which are executed by the 'winning' rule. Such systems have been used to control a robot in simulation, for example [18] and also animats navigating in environments containing aliasing states, for example [8]. However, although these system generate control rules automatically, individual rules are distinct and there is no interaction between rules, therefore a pure classifier system approach cannot represent sequences of actions which is essential if the goals of this research are to be met. However, both the work of [18] and [8] partially informs the architecture proposed here, in particular in the chosen representation of antibodies in the network with regard to representing sensor information and motor actions.

Finally, [2] proposes a developmental mechanism which has some similarities to the proposed method, but it is not clear whether his system is scalable. His work, and its relation to our proposed model, is further discussed in section 3.1.

3 A New Architecture

Let us suppose that at the very start, the robot is driven by basic instincts such as a 'desire to avoid collisions' and a 'desire to seek novelty'. The robot should learn through experience, and the learned behaviours should gradually take over control from the instinct-driven initial system. The robot therefore needs to capture some minimal details of its experiences. In the proposed model, depicted in figure 1, this information is held as a collection of *rule-like associations* (RLAs). Each RLA is a node in a network and consists of a (partial) description C of sensory information, a robot action command A and a partial description of the sensory effects E of doing the action. After creation, an RLA therefore expresses some of the expected results of doing action A in a context C, and weighted network links express the sequencing information; a sub-path involving strongly positive weights would express an episode.

In immunological terminology, antibodies correspond to these RLAs, and antigens correspond to sensory data (not necessarily just raw data, see below); the C and E parts of an RLA can be regarded as paratope and epitope. Much as in Jerne's [7] immune-network hypothesis, connections are formed and adjusted by a process of *recognition* between the paratope of one antibody and the epitope of another, and result in stimulation and suppression of one antibody by another, according to a dynamical equation of the form given in equation 1, as first suggested by Farmer in [3]. In this equation, $a_i(t) \geq 0$ represents the strength or concentration of antibody i at time t, e_i represents the stimulation of antibody i by the antigen (current sensory information), the first summation term represents the total stimulation of the antibody i from the other antibodies in the network, the second summation term represents the suppression of antibody i from other antibodies in the network, and k_i is a natural decay factor.

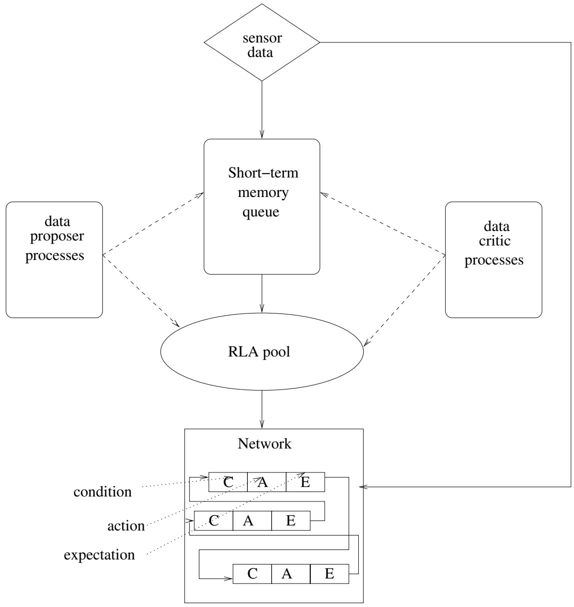

Fig. 1. A schematic representation of the proposed architecture

$$\frac{da_i(t)}{dt} = \left(e_i + \sum_{j=1}^{N} m_{ij} a_j(t) - \sum_{j=1}^{N} m'_{ji} a_j(t) - k_i \right) a_i(t) \tag{1}$$

Immune system models require a mechanism by which recognition can occur. For example, AIS network models (e.g. [16]) often use the Euclidean distance in data-space between two data-items to signify recognition. In the proposed architecture, recognition between RLAs or antibodies serves the following purposes:

– individually, they can express a temporal association between RLAs – a strong positive connection between X and Y means that if RLA X fits the current situation then RLA Y is a possible candidate to describe the subsequent situation. Thus, individually, they can capture some aspects of episodic memory. Importantly, the boundaries of episodes can emerge from the dynamics of the network. That is, an episode ends when there is no clear winner as to the successor RLA.
– individually, they can (as inhibitory links) express a competition between different RLAs to account for the current situation.
– collectively, they can act as an attentional mechanism. The dynamics of the network can cause it to settle to a state in which some set of RLAs are reasonably active and the remainder are not; the active set represents the 'current memory context', as it were. In the set of linear differential equations in equation 1 above, the system can only have either a point attractor or a limit cycle depending on the values of the constants involved, but note that

the e_i would normally be time-dependent (the external data changes with robot activity) so the system should be capable of flipping between different attractors and limit cycles as the robot moves and the environment changes.

Clearly, this approach raises several questions. By what process(es) are RLAs created, and on what timescales? How are the connections between antibodies formed, and the strength of their affinities quantified? How will the dynamics of the network operate? We consider these questions next.

3.1 Generating RLAs

Although the aim of the architecture is to provide a framework in which the robot can grow-up, it seems reasonable to start with a system that has built-in basic behaviours, for example “explore”, “avoid obstacles”, “avoid boredom”. We propose that this is handled by a partially pre-built network of RLAs, which then undergoes adaptation and growth until it becomes capable of allowing the robot to perform non-trivial and purposeful-seeming sequences of actions.

First, there is a short fixed-length queue that contains recent interesting sensory and motor events. The queue provides a form of short-term or working memory and distantly resembles human short-term memory which experimental studies have suggested is of bounded capacity (although expandable through lengthy training) and contains things that are fairly closely linked to the sensory input. For our initial purposes, ‘interesting’ means ‘significantly changing’; for example if the robot is moving straight ahead across a vast empty space, the queue should not alter. The contents of this queue provide the raw material from which candidate RLAs can be built and then inserted into the network. Clearly the queue needs to contain some consequences of an action before this can happen, so RLAs can only get created at certain moments. The RLA pool can be viewed as containing fragments of experience. We propose RLAs of the following form:

```
RLA-3:
   condition: front-sensors = high
              and left-sensors = low
              and moving-average-of-front-sensors = low
      action: turn right
 expectation: left-sensors = high
  activation: 0.05
       links: 7/0.9, 453/-0.2, 64/1.2
```

Note that the condition does not fully describe the raw sensor data, and may refer to higher-level data constructs at later stages of the robots development. At the very start only raw sensor data will be available, but in real application, this will contain far too much information to be useful. So, abstractions will be proposed – for example, natural ones to suggest at the start would contain either thresholded or thresholded-moving-average versions of raw sensory information. We envisage that there will be some *data proposer and data critic processes* that suggest and evaluate new data abstractions built out of all existing data items (whether raw or already abstracted). Thus, the data universe will be dynamic. It is envisaged that the RLA proposer processes will gradually generate RLAs representing higher and higher levels of knowledge, thus representing the robot 'growing-up' in terms of its capabilities to understand its world. Thus, for example, an early set of RLAs composed of raw sensor data indicating that a robots left and front sensors are high, might eventually be replaced by an RLA representing the concept 'corner', with an associated action to turn right. Note that this thinking has some similarities with the work of Drescher [2] who introduced a general learning and concept-building mechanism called the *schema mechanism* in order to reproduce aspects of Piagetian cognitive development [11] during infancy. In this mechanism, the world is initially represented only in terms of very simple motor and sensor elements. Crucially however, the mechanism can define new, abstract, actions and invent novel concepts by constructing new state elements to describe aspects of the world that the existing repertoire of representations fails to express. Eventually, representations are discovered which can represent an object independently of how it is currently perceived and may be far removed from the original description.

Newly-formed RLAs will be presented to the network, where they will survive by being found to be useful and continue to survive only by continuing to be useful. Conversely, RLAs will be removed from the network if their stimulation falls below some threshold value. Data proposer processes are likely to be based on clustering techniques, for example k-means clustering or self-organising maps. Recent work by Prem *et al* in relation to this architecture shows promising results in using the ISO-map technique [12] for finding abstractions in time-series of sensor data generated by a real-robot. Data critic processes are likely to be based on checking whether data items have become redundant.

3.2 Quantifying Recognition between RLAs

As already stated, there is no straightforward way of quantifying the extent to which one RLA in the network should recognise another. As already mentioned in section 2, [10,17] tackled this problem by using a genetic algorithm, but this method has significant disadvantages if the goals of the new architecture are to be achieved. Firstly, use of a GA is likely to be too computationally expensive and slow in a real robotic environment, and furthermore, the connection strengths between antibodies could possibly change over time as the robot learns more about its environment, which would require the use of a continuously running GA. This type of process does not really have an analogy in the biological immune system in which connection strengths are determined by physical binding processes which do not alter over time, but there is an obvious analogy with the kind of Hebbian learning processes occurring in neural networks in which connection strengths are continuously adjusted over time.

However, [5] describes use an on-line adaption mechanism in an immune-network for achieving behaviour-arbitration – in this mechanism, affinity values are adaptively modified until the required behaviour emerges. This type of approach familiar to reinforcement learning appears to be more promising when using real robots, and hence will be adopted in this architecture.

3.3 Network Dynamics

As mentioned in section 3.1, the RLA pool can be thought of as containing fragments of experience which may become incorporated into the network. Initially, the network should consist of instinct driven behaviours but over time, these should be replaced by more sophisticated behaviours – however, it seems reasonable that the network should still maintain some record of these instinctive behaviours, as they may be useful at points in the future, and hence can override other behaviours given the right conditions.

Biological and neurological studies tell us that the network cannot be infinitely large; the brain has a finite volume in which neurons can exist, and similarly the immune system cannot physically contain an infinite number of antibodies (and anyway, the number of different types of antibodies is limited by the diversity of the DNA from which they can be formed) hence it seems logical and practical that the size of the network must somehow be bounded. Various mechanisms for achieving this can be found in the literature; plausible ones would seem to be based on the notion of a competition for resources, where RLAs would have to prove their worth to be allowed to remain in the network else be replaced by others. The natural decay constant k_i of the antibody would aid this process but further 'cell-death' mechanisms need to be investigated.

3.4 The Emergence of Planning

Planning-like behaviour should emerge from the network: this could occur as a dynamic cascade of internal events. For example, a goal is represented as an antigen which is injected into the system. As in the immunological system, the network must respond to this antigen - the antigen (goal) remains in the system until it is satisfied. At any point in time, the external environment will consist of multiple and changing data items, representing goals, sensory information and (perhaps) maps and internal memory states; the resulting course of action is results from a chain of RLAs firing, determined by the dynamically changing concentrations of the antibodies. Thus, the network effectively records chains of events that can allow a desired goal to be achieved. This may lead to the emergence of more complex behaviours.

Alternatively, a more classical planning approach could be taken. The RLAs associate expectations with states, therefore in theory a *virtual* antigen could be injected into the system, representing some potential goal or action, and the dynamical equations applied to determine what would be the result of such an action. By comparing the results of a number of such virtual experiments, a 'plan' could then be selected. The network thus provides a blackboard for 'thought' experiments by the robot.

4 An Initial, Partial Implementation of the Ideas

This section describes an experiment performed as a proof of concept for the architecture, though clearly it is only a basic skeleton of the proposed system. We used Olivier Michel's simulation (http://diwww.epfl.ch/lami/team/michel/khep-sim/) of a Khepera robot. This robot has six forward-looking IR sensors and two backward-looking ones. Each returns a value between 0 (nothing sensed) and 1023 (object very close), but disturbed by significant noise. The robot has two wheels controlled by stepped motors, each wheel can be commanded to go forward or backward, by an integer amount in the range 0 to 10. The robot's world is bounded and contains user-configurable internal obstacles. We used just the default world for this experiment and the robot actions were limited to moving forward at speed 4 (ie both motors), or turning left or right by 45 or by 90 degrees – that is, five possible actions.

In order to perform a proof-of-concept demonstration, a set of 32 *hand-crafted* RLAs were produced, using the representation described below. It should be emphasised however that in the final system all RLAs should be generated automatically by the system – methods for achieving this are currently under investigation. The concentrations of each antibody were initially all set to a value of 0.1, and initially, there were no links in the network (that is, all link weights were 0).

As previously mentioned, antigens should capture the essence of the current sensory experience of the robot. In this initial model, an antigen consists of a binary string representing 2 types of sensory information; the first captures the current sensory data, the second attempts to maintain some record of the recent history of the robots experience. In both cases, rather than deal only with raw sensory data, we describe the sensory information in a binary string of total length 24 bits. The first 8 bits represent thresholded sensor values from each of the robots eight sensors (1=over the threshold, 0=under the threshold). A moving average of each sensor value is also maintained over 5 timesteps, and each of these values is converted into a 2-bit value (00=0-255, 01=256-511, 10=512-767, 11=768-1023) resulting in a further 16 bits.

Antibodies consisted of a binary string with 3 parts. The first 24 bits representing the condition part of the RLA corresponds to the current sensory information, and thus has the same form as the antigens. The 2nd part denotes a motor action, and the final part, the paratope, represents the expectation of the sensory conditions that should prevail following execution of the action, and therefore again consists of 24 bits. In this case however, the bits can contain 'don't care' symbols.

The algorithm described below was run over a period of 100,000 timesteps which took about 9 minutes using the Khepera simulator on a 1GHz PC.

1. Initialise a pool of antibodies with a concentration of 0.1. At this stage, all connections have a strength of 0.0
2. At time t:
   - Present an antigen representing sensory current conditions to the network

  - Apply equation 1 to update the concentration of all antibodies in the network
  - Select the antibody with highest concentration and execute its action - call this antibody x
3. At time $t + 1$:
  - Get the current antigen
  - If the match between the current antigen and the expectation of the previously selected antibody x is greater than some threshold, update the links between x and all antibodies whose condition part matches the expectation of x, by an amount δ_1 proportional to the strength of the match
  - *else* if the expectation was incorrect, decrease the strength of all links emanating from x by an amount δ_2.
4. Goto step (1)

The matching algorithm is simple; it does a bit-by-bit comparison and accumulates a 'match score'. A bit-comparison involing a 'don't care' scores +1; if there is no 'don't care', then equality is worth +2 and inequality is worth −2. Thus the score can range between −48 and +48, and the threshold we use is 80% of maximum. In this experiment, we chose $\delta_1 = \delta_2 = 0.01$ but this clearly influences stability and speed of adaptability, and needs further experiment.

4.1 Results

After running the algorithm for an initial learning phase of 100000 iterations, the RLAs chosen by the algorithm were recorded. A flow diagram showing the sequences of surviving RLAs is shown in figure 2. Note that this is not the immune network topology, but is instead used to illustrate how the different sensory experiences are captured when using the immune network approach. As an example, the RLA sequence 0, 1, 5, 6, 8, 0 captures the sequence of events that occur when the robot meets an obstacle head-on and turns to avoid it. This sequence could be interpreted as: 'sensing clear space, go forward, obstacle looming in front, go forward, obstacle ahead, turn right, obstacle to the left, turn right, obstacle more to the left, turn right, sensing clear space, go forward' and so on.

The remaining RLAs appear to capture some episodes (sequences of sensory events) in a reasonably stable manner, thus the robot could be said to have a long-term memory that maintains a record of the relationships between sensory situations, actions performed and the effects of those actions.

5 Conclusion

In this paper we have proposed a robot control architecture based on an AIS that should be capable of capturing at least some aspects of 'growing up' through experience. An initial experiment showed that it seemed to be capable of capturing some episodes of experience. However, a lot more remains to be done and

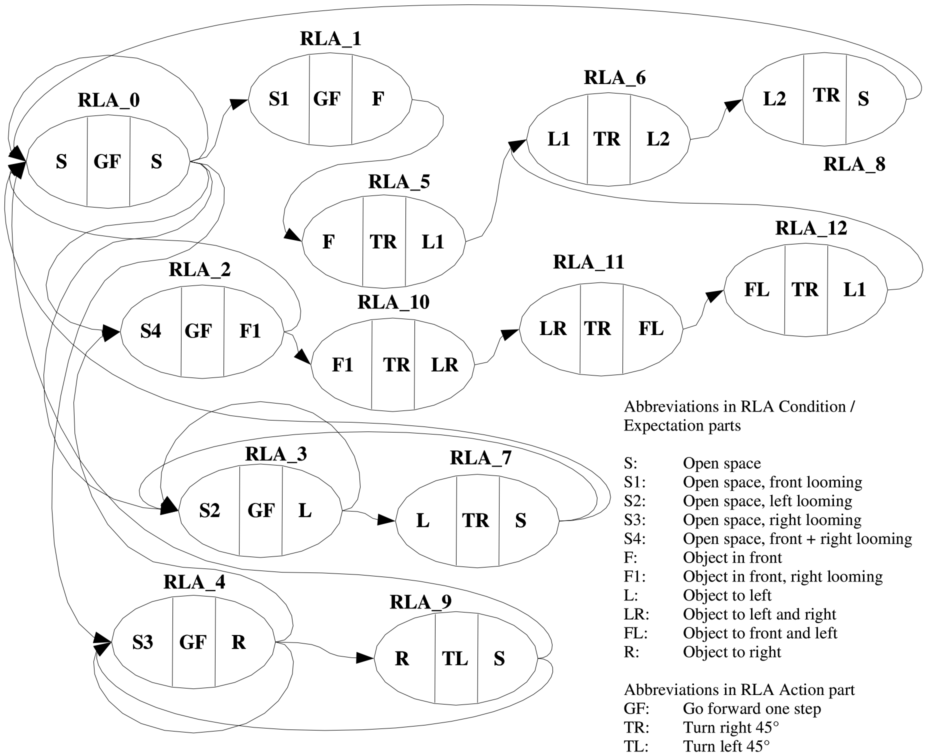

Fig. 2. Flow diagram showing sequences of chosen RLAs

in particular we have not yet done any experimental investigation of the idea of changing the data universe dynamically, nor have we done much exploration of the sensitivity of the system to the many choices involved. Clearly, much work also needs to be performed in investigating the scalability of the system. Furthermore, it is well known in robotics research that simulated systems rarely transfer seamlessly to the real-world, therefore we fully intend to transfer this architecture to a real-robot (see [6]).

However, we do believe that what we have sketched out represents a very fruitful line of work, both in terms of studying robot development and in terms of studying AISs. Too much research in AISs still relies on overly-simplistic metaphors. We claim that the problems of robot development provide an excellent context for studying AIS issues such as sophisticated matching algorithms, the dynamics of network models, the problems of handling a continually-evolving representation and even the computational tractability of AISs.

Acknowledgements

This work is supported by the European Union funded IST programme, grant no. IST-2000-29225.

References

1. Anonymous for reviewing purposes. Requirements for getting a robot to grow-up. In *Proceedings of the European Conference on Artificial Life, ECAL 2003*, 2003.
2. G. Drescher. *Made-Up Minds: A Constructivist Approach to Artificial Intelligence.* MIT Press, 1991.
3. J.D Farmer, H Packard N, and A.H Perelson. The immune system, adaption and machine learning. *Physica D*, 22:187–204, 1986.
4. A. Ishiguro, T. Kondo, Y. Watanabe, Y. Shirai, and H. Uchikawa. Immunoid: a robot with a decentralized consensus-making mechanism based on the immune system. In *Proceedings of ICMAS Workshop on Immunity-based Systems*, pages 82–92, 1996.
5. A. Ishiguro, S. Kuboshiki, S. Ichikawa, and Y. Uchikawa. Gait-control of hexapod walking robots using mutual coupled immune networks. *Advanced Robotics*, 10(2):179–196, 1996.
6. IST-2000-29225. Systemic intelligence for growing up artefacts that live. http://www.ist-signal.org.
7. N.K Jerne. The immune system. *Scientific American*, 229(1):52:60, 1973.
8. Pier Luca Lanzi. An analysis of the memory mechanism of XCSM. In *Genetic Programming 1998: Proceedings of the Third Annual Confere nce*, pages 643–65. Morgan Kaufman, 1998.
9. D. Lee, H. Jun, and K. Sim. Artifical immune system for realisation of cooperative strategies and group behaviour in collective autonomous mobile robots. In *Proceedings of Fourth Internation Symposium on Artifical Life and Robotics*, pages 232–235, 1999.
10. R. Michelan and F.J. von Zuben. Decentralized control system for autonomous navigation based on an evolved artificial immune system. In C.L. Giles et al, editor, *Proceedings of CEC-02*, pages 1021–1026. IEEE Press, 2002.
11. J. Piaget. *The Origins of Intelligence in Children.* Norton, N.Y., 1952.
12. E. Prem and P. Poelz. Concept acquistion using isomap on sensorimotor experiences of a mobile robot. In *Proceedings of International Workshop on Epigenetic Robotics*, Boston, August 2003. to appear.
13. Surya Singh and Scott Thayer. Immunology directed methods for distributed robotics: A novel, immunity-based architecture for robust control and coordination. In *Proceedings of SPIE: Mobile Robots XVI*, volume 4573, November 2001.
14. Surya Singh and Scott Thayer. A foundation for kilorobotic exploration. In *Proceedings of the Congress on Evolutionary Computation at the 2002 IEEE World Congress on Computational Intelligence*, May 2002.
15. Surya Singh and Scott Thayer. Kilorobot search and rescue using an immunologically inspired approach. In *Distributed Autonomous Robotic Systems*, volume 5. Springer-Verlag, June 2002.
16. Jon Timmis and Mark Neal. A resource limited artificial immune system for data analysis. *Knowledge Based Systems*, 14(3-4):121–130, June 2001.
17. Y. Watanabe, A. Ishiguro, Y. Shiraio, and Y. Uchikawa. Emergent construction of behaviour arbitration mechanism based on the immune system. *Advanced Robotics*, 12(3):227–242, 1998.
18. A. Webb, E. Hart, P. Ross, and A. Lawson. Controlling a simulated khepera with an xcs classifier system. In *Proceedings of 7th European Conference on A-Life.* Springer-Verlag, 2003. to appear.