Critical Damage Reporting in Intelligent Sensor Networks

Jiaming Li, Ying Guo, and Geoff Poulton

Intelligent Systems,
CSIRO Information and Communication Technology Centre,
Marsfield NSW 2122, Australia
{Jiaming.Li, Ying.Guo, Geoff.Poulton}@csiro.au

Abstract. In this paper, we present a Top-Down/Bottom-Up (TDBU) design approach for critical damage reporting in intelligent sensor networks. This approach is a minimal hierarchical decomposition of the problem, which seeks a balance between achievability and complexity. Our simulated environment models two-dimensional square cells as autonomous agents which sense their local environment, reporting critical damage as rapidly as possible to a report delivery site (portal) by using only the adjacent-cell communication links. The global goal is to design agent properties which will allow the multi-agent network to detect critical damage anywhere on the network and to communicate this information to a portal whose location is unknown to the agents. We apply a TDBU approach together with genetic algorithms (GA) to address the global goal. Simulations show that our system can successfully report critical damage much better than random methods.

1 Introduction

Intelligent sensor networks have been investigated recently for a number of applications including structural health monitoring, which is a critical factor for future aerospace vehicles. Such vehicles must operate in adverse environments where failure to recognise, assess and respond adequately to damage may prove disastrous. The advantage of intelligent sensor networks in such environments lies in the distributed nature of the intelligence which allows the monitoring process to continue even when considerable damage exists. This situation is far more robust than a more conventional centralised intelligence where damage to the central processor may disable the entire system [1].

The Ageless Aerospace Vehicle (AAV) project is being conducted jointly by CSIRO and NASA with the aim of investigating the use of intelligent sensor networks for structural health monitoring of future aerospace vehicles [2]. As part of this project a Concept Demonstrator (CD) system has been developed. Shown in Fig. 1, the CD is a hexagonal structure of approximately 1m. diameter and 1m. in length, covered by 48 1mm. thick aluminium panels behind which is a rectangular array of 192

G.I. Webb and Xinghuo Yu (Eds.): AI 2004, LNAI 3339, pp. 26–38, 2004.

sensor cells, each having four piezoelectric sensors and a microprocessor. Each cell also has the ability to communicate only with its four immediate neighbours. Since the cells have sensing and acting capabilities and are imbued with independent intelligence they may be regarded as "agents", and the sensor network itself is an example of a multi-agent system.

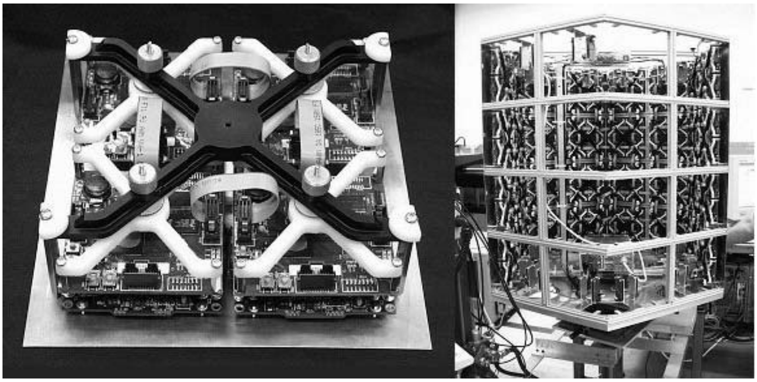


Fig. 1. Ageless Aerospace Vehicle Concept Demonstrator, showing an aluminium panel with four cells (left) and the demonstrator with four of the six sides populated (right)

This sensor network is intended to detect and assess impacts on the skin from fast-moving projectiles (which simulate micrometeoroids in a space environment). The degree of intelligence of the network can be varied by programming the microprocessors. A number of different detection, assessment and reporting tasks are possible, including determination of the location and severity of impacts together with an assessment of damage, both immediate and cumulative. Eventually prognosis of the effects of damage on the fitness of the vehicle and the ability to self-repair are envisaged. Although the network will have no control centre it will generally be the case that communication from a damage site to another part of the vehicle will be required, for example to initiate secondary inspections, repair or, in extreme cases, appropriate emergency action. Such communications will most likely be hierarchical and flexible, so that the report delivery site (portal) will vary with time as well as depending on where the damage occurred and its severity.

This paper examines the reporting of critical damage in such intelligent sensor networks. "Critical damage" means an impact severe enough to threaten the survival of the vehicle. In such situations time is of the essence, and the requirements on the network are to send an alarm as rapidly as possible to a (probably) unknown location using only the adjacent-cell communication links. In addition, there may exist barriers to communication due to the network configuration itself or to significant prior and continuing damage. Thus the communications environment is also unknown and changing.

The multi-agent sensor network described above is likely to be a complex system exhibiting emergent behaviour [3]. Such systems make life difficult for the designer, principally because of the essential unpredictability of emergence. On the other hand, emergence can offer a much richer solution space and lead to better solutions if the unpredictability can be controlled. Biological evolution offers many examples where this has been used to advantage [4]. A traditional approach to the design of complex systems is hierarchical decomposition [5], where the problem is broken down into a (sometimes large) number of layers which are more amenable to solution. Unfortunately this process, whilst often allowing a design to be achieved almost always suppresses emergent behaviour, thus denying the designer access to the rich solution space which complexity can provide. The authors have recently introduced an alternative to this approach which gives the advantages of hierarchical decomposition whilst retaining the possibility of emergent behaviour [6, 7]. Called Top-Down/Bottom-Up (TDBU) design, it is really a minimal decomposition of the problem which seeks to retain the complex nature of the original. The TDBU approach is described in more detail in the next section.

2 Top-Down/Bottom-Up (TDBU) Design

Our approach is to seek a balance between "top-down" (engineering) and "bottom-up" (scientific) processes. Engineering design starts with a system goal and employs a top-down approach to formulate more achievable intermediate goals. In contrast, the scientific method develops new knowledge of what is achievable by working from the bottom-up. Successful design is possible when the two processes can be matched, with intermediate "entities" (engineering goals) being capable of being achieved using existing scientific understanding. To access the rich space of potential solutions available from complex systems, it is important to preserve emergent behaviours that would be lost with a fully hierarchical design. A minimal hierarchical decomposition is a means of seeking a balance between achievability and complexity.

Of course, it is possible to bypass the TDBU process by using a genetic algorithm (GA) or similar to design directly for a specific goal. The disadvantages, however, are lack of generalisability and having to repeat time-consuming GAs for each design.

In contrast the TDBU approach can retain emergence in one (or both) parts, thus broadening the solution space. This is possible because, although the "intermediate entities" may result from an emergent process, they may be usable as generic building blocks to achieve a broader range of goals in the solution space, possibly leading to general design rules. Also, splitting the problem will lead to simpler optimization in most cases.

3 Application to the Sensor Network

3.1 Environment and Assumptions

A simplified version of a sensor network is a W x H array of squares, with each square representing a cell (agent) of the network. All agents are assumed to have identical properties which will be discussed in more detail below. One (or more) of the agents is designated as a “portal”, the location to which, at any given time, critical damage must be reported. Any agent may become a portal and indeed the portal location may vary with time and circumstance. The rules for selecting the portal fall outside the scope of the present study, but may be due to decisions made outside the sensor network, or alternatively may be part of a self-organised hierarchical process which is an emergent property of the network itself [8]. The network may contain barriers to communication, across which communication cannot take place. Barriers may be inherent in the structure or due to prior damage. An example of such a network is shown in Figure 3.

3.1.1 Agent Properties

- Each agent may communicate directly with its four neighbours. For the purposes of this study two levels of communication will be defined: (a) Status query, a continuing process whereby each agent periodically makes and responds to status requests of its neighbours. Failure to respond (or a fault-indicating response) will set a flag which, after consistency checks, results in the initiation of a critical damage report. (b) Normal reporting, where an agent transmits a message to one or more of its neighbours.
- An agent has memory and can store data such as state, signals, IDs, logic, action lists, parameters, or programs.
- An agent has the ability to perform calculations.
- Each agent can become a portal, and the above resources must be sufficient to allow this.

3.2 Objective - Critical Damage Reporting

The objective is to design agent properties to allow detection of critical damage anywhere on the network and to communicate this information to a portal. The portal, which may be any agent in the network, is assumed to be capable of transferring the message to another part of the system which is capable of taking appropriate action. Critical damage is defined in this instance by the failure of an agent to respond to periodic status queries from a neighbour. Information about the location and severity of damage will not be reported in this initial trial, just the fact that critical damage has occurred. Time is of the essence for critical damage, so successful reporting in minimum time is the aim, over a wide variety of damage and environmental conditions. Minimising the use of resources (communications, etc.) is a subsidiary goal.

Agents have local knowledge only, so portal location and network status are unknown to them. Initial trials will assume a single portal only. A TDBU approach together with genetic algorithms will be used to design the required agent parameters.

3.3 TDBU Design for Critical Damage Reporting

This design problem can readily be made to match the TDBU framework by recognising that agents with portal status act differently to other agents. The problem may then be split into two parts, dealing with normal and portal agents respectively.

This method of splitting the problem has advantages since it is very likely that good solutions will involve communications from the portal as well as from the neighbours of a damaged cell. The value of this is that the portal may pass messages from cell to cell, storing the direction back to itself in each cell. This establishes a network of cells which know the portal direction and can guide incoming critical damage messages. Any such network must be able to operate effectively if portal positions change periodically by reconfiguring itself to accommodate such changes. One way of achieving this is to allow stored information about the portal to decay at a rate dependent on how much the portal moves.

This process has much in common with well known "ant" algorithms and the "pheromones" with which their tracks are marked [9]. However, we do not wish to restrict message-passing or the storage of information to single-track ant-like behaviour since there may be other types of behaviour which give better results. However, we will make use of the ant/pheromone terminology in the interests of brevity.

The top-down part of the design is thus provided by the variety of pheromone networks that may be generated by the portal. The intermediate entities may be identified with the pheromone networks themselves.

The corresponding bottom-up part of the design is to communicate critical damage to the portal with its pheromone network. In general this will be an easier task than locating the portal itself since use can be made of the information in the network about portal location.

The design space covers the reporting of critical damage over a wide range of conditions, including the existing prior damage/barrier environment, the rate of new damage and portal mobility.

3.4 Design Assumptions

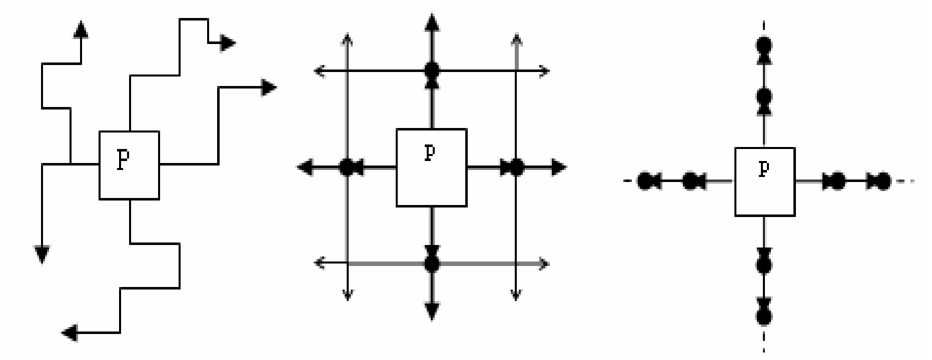
The design is accomplished by finding sets of agent parameters which best provide time-critical reporting of damage for a wide range of environmental conditions, including variation of the rate of damage and the presence of boundaries. Having used the TDBU process to split the problem as described above we will proceed as follows:

(a) Agent behaviour when acting as a portal will be specified by the designer.
(b) A genetic algorithm (GA) will be used to design optimum agent parameters for a given fitness function, for agents in damage report mode.
(c) The overall solution will be tested on various environmental conditions and decisions made about the regions of applicability.
(d) The process will be repeated for other fitness functions and portal properties.

In each case the performance will be compared with benchmarks, including the case where agent properties provide random communications for damage reporting and no portal communications at all.

3.4.1 Examples of Portal Pheromone Networks

There are many choices for the type of networks of pheromones set up by portals, ranging from no network at all to a “flooding” broadcast. Figure 2 illustrates some of those used in the present simulations.



(a) Random (b) Broadcast (c) Cross

Fig. 2. Some examples of portal’s information (pheromone) distribution

4 Simulation and Results

4.1 Simulation Environment and Agents’ Properties

A simulation environment like that shown in Figure 3 was set up to test agent design in a TDBU framework. A “damage report” event has randomly located portal and damage sites, with both the portal and neighbours of the damaged agent being capable of sending messages. It is assumed that each event causes all four neighbours of a damaged cell to immediately initiate messages which are propagated through the network according to the agent properties. The aim is to find agent properties for maximum robustness of reporting to the portal and minimum reporting time, averaged over many events. Minimising communications cost is a subsidiary goal. Two key functions, G and G_p, determine how an agent responds to a message from a damage site or a portal respectively, telling it whether and in which direction to pass the message on. The parameters of G and G_p define the agent properties and are thus what must be found to optimise the performance of the network.

In these tests the portal-derived function G_p will be pre-chosen by the designer from a list of choices. The damage-derived function G, on the other hand, will be optimised using a genetic algorithm (GA).

The details of the simulation environment, agent properties and the genetic algorithm are as follows.

4.1.1 Simulation Environment

The simulation environment is a two-dimensional array as in Figure 3, with width $W = 10$ and height $H = 10$ cells. All agents are identical. For some tests a barrier to communications occupies the cells (5,1) to (5,5).

4.1.2 Agent Parameters and Properties

4.1.2.1 Data Storage

An agent stores the following data, if available:

- V_p: pheromone value (default zero);
- P: pheromone direction, where the portal-derived signal comes from;
- D: damage direction, where the damaged-derived signal comes from;
- g_{pr}: reporting direction from the last previous event (if it exists).
- r: a random direction generated by a reporting event.

V_p takes on integer values from zero to a maximum value V_{max}.

The domain of P, D *and* r is {UP, DOWN, LEFT, RIGHT, NONE}, ie. the four directions plus the option of not sending a message.

4.1.2.2 The Damage-Derived Function G

There are many choices for this function, which has the following form:

$$g = G(P, D, g_{pr}, r; \mathbf{w}),$$

where g is the direction of the outgoing message and $\mathbf{w} = [w_P, w_D, w_g, w_r]$ is a vector of weights associated with the four other parameters. Note that for these tests g depends on the direction, but not the value, of any pheromone present. The weights are real numbers in the domain [0, 1].

The choice of G for the present tests may be described as follows.

1. Define a vector $\mathbf{v} = [v_U\ v_D\ v_L\ v_R]$, where v_U, v_D, v_L, and v_R are the weights of the directions UP, DOWN, LEFT, RIGHT respectively. $\mathbf{v}$ is initially [0 0 0 0].
2. $\mathbf{v}$ is updated based on the values of the parameters P, D, g_{pr}, or r and their associated weights. For example, if P is “UP”, v_U is updated by adding the weight of P, w_P: $v_U = v_U + w_P$. This process is carried out for all parameters.
3. Finally, the report direction g corresponds to the maximum element of $\mathbf{v}$.

A benchmark report-direction function G_{B1} was also used to test the efficacy of G. This is simply defined $g = r$, representing a fully random report direction.

4.1.2.3 The Portal-Derived Functions G_P and G_{PV}

The general form of the portal-derived function is similar to that for G, except that the pheremone value V_P needs to be considered. This requires an additional function to describe what pheremone value is passed. Thus,

$$g = G_P(V_P, P, D, g_{pr}, r; \mathbf{w}_P), \qquad V_{Pnew} = G_{PV}(V_P, P, D, g_{pr}, r; \mathbf{w}_P),$$

Again, g is the direction of the outgoing message and $V_{P_{new}}$ is the pheromone value passed to the next cell. The weight $\mathbf{w}_P$ has the same domain as $\mathbf{w}$.

As mentioned above, user-selected portal-derived functions will be in our tests. In all cases the pheromone value will decrease by one as it passes from agent to agent, ie. $V_{Pnew} = V_P - 1$.

Four examples of G_P are used and are described below. Because of their simplicity explicit functional representation is unnecessary.

- G_{P1} (Null): No portal-derived messages are passed on;
- G_{P2} (Ants): The signal direction depends only on the damage direction (75%) with a 25% random component;
- G_{P3} (Broadcast): The portal-derived signal is sent to all possible directions;
- G_{P4} (Cross): Signals maintain their original direction up to the edges of the environment.

4.1.2.4 Pheromone Decay and Portal Properties

Two other agent properties need mentioning. (i) Pheromone decay, where an agent's pheromone value decreases with time according to a specified rule. Only two situations are used in the current tests: (a) No decay, where pheromone values do not decrease, and (b) linear decrement, where the value decreases linearly.

The second is the behaviour of the portal which, although it is an agent like any other, has special properties, principally the initiation of messages. For simplicity the following assumptions have been made regarding a portal.

1. The portal issues messages from all four ports at once (or none at all).
2. When an agent becomes a portal it assumes the maximum pheromone vale V_{max}, which may decrease with time if pheromone decay operates.
3. When the pheromone value of a portal reaches a given threshold it issues new messages from all its ports.

4.2 Genetic Algorithm-Based Design

Genetic Algorithms (GA) are robust in complex search spaces and are appropriate for our current situation [10]. A colony of individuals (parameter sets) can be evolved for a number of generations, improving the performance of the colony. At the end of each generation, "parent" individuals are selected based on a fitness computation which must be strongly related to the desired outcome. After the two parents are selected, each is represented by a "chromosomal" string and are then combined, using one of several methods, to form two new chromosomes. Some old individuals are then replaced in the colony by the offspring (cf. [10, 11] for a detailed description of such algorithms).

In the current tests GA is used to optimise the parameters of the report-direction function G. In every generation each individual of the colony must be evaluated by calculating a value using an appropriate "fitness function", which is a well-behaved measure of relative fitness. For the current problem, obviously individuals that can robustly report damage to the portal in minimum time and with low communication costs will score highly. We define several such functions as follows.

$$f_1 = \sum_{i=1}^{N}\sum_{j=1}^{M} S_{i,j}\,, \quad S_{i,j} = \begin{cases} 1 & \text{if message reaches portal} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

Where f_1 is the fitness value, N is repeat time for each individual in the GA, M is the number of neighbours for each event. This fitness function only judges whether the damage report has arrived at the portal or not. Time and communications costs are ignored.

$$f_2 = \sum_{i=1}^{N}\sum_{j=1}^{M} \Psi_{i,j}\,, \quad \Psi_{i,j} = \begin{cases} -t_{i,j} & \text{if message reaches portal} \\ -2000 & \text{otherwise} \end{cases} \tag{2}$$

A large penalty is given for any signals not reaching the portal. Compared with equation (1), this fitness function includes the reporting efficiency, where t is the reporting time that the neighbour has taken.

$$f_3 = \sum_{i=1}^{N} \min_j C_{i,j}\,, \quad C_{i,j} = \begin{cases} -t_i & \text{at least one neighbour contacts portal} \\ -2000 & \text{otherwise} \end{cases} \tag{3}$$

f_3 differs from f_2 in that there is no penalty as long as at least one neighbour reports the damage to the portal.

$$f_4 = \sum_{i=1}^{N} \frac{C_i}{\tau_i}\,. \tag{4}$$

This fitness value is normalized by the minimum possible time for the neighbour to report to the portal, τ_i.

4.3 Experimental Results and Comparison

Two groups of experiments were conducted: (1) No barriers and no pheromone decay; (2) A single obstacle as in Figure 3 and pheromones decaying at constant rate.

Two functions help in judging aspects of the performance. The first is success rate,

$$S_R = \frac{S}{L}\,. \tag{5}$$

Where S is the number of successful reports and L the number of events. A report is successful if at least one message per event reaches the portal.

Secondly, the report efficiency is defined as follows.

$$E = \frac{1}{L}\sum_{k=1}^{L} \frac{\sum \tau_{k,i}}{\sum t_{k,i}}\,. \tag{6}$$

Where $\tau_{k,i}$ is the minimum possible time for a message to reach the portal, and $t_{k,i}$ the actual time for message *i* in event *k*.

4.3.1 No Obstacle and No Pheromone Decay

With no barriers, GA used to find the optimum damage-derived parameters. The highest pheromone value was V_{max} =8. GA training was very similar to that described in [2]. After training the performance of the best individuals was measured in terms of success rate and efficiency. Table 1 lists the test results using different strategies described in section 4.1. Each test result is the average over 1000 repeats.

Table 1. GA results comparison

	Portal-derived Function	Damage-derived Function	Fitness functions	Initial Generation Performance	Best Individual Performance	
				S_R	S_R	E
Test1	$\boldsymbol{G_{P1}}$	$\boldsymbol{G_{B1}}$	n.a.	n.a.	1%	2.3%
Test2	$\boldsymbol{G_{P2}}$	$\boldsymbol{G}$	Eq.(1)	27%	100%	18.40%
Test3	$\boldsymbol{G_{P2}}$	$\boldsymbol{G}$	Eq.(2)	27%	100%	46.92%
Test4	$\boldsymbol{G_{P2}}$	$\boldsymbol{G}$	Eq.(3)	27%	100%	47.12%
Test5	$\boldsymbol{G_{P2}}$	$\boldsymbol{G}$	Eq.(4)	27%	100%	47.55%

From Table 1 we can see that the damage-derived function is the main factor that affects the report efficiency. The benchmark results were far worse than any of the GA-designed results. Since fitness function f_1 did not consider time cost, Test2 is worse than Test3 to Test5 in report efficiency. f_2, f_3 and f_4 have little difference for either measure. We may conclude that the damage-derived function $\boldsymbol{G}$, found by GA, is a good design for any of these fitness functions. From the Table 1 we can also see the initial generations' performances are much worse than the best individuals found by GA after the evolution process.

Table 2. Performance of best strategies for other portal rules

Strategy	Portal Rule	S_R	E
Test5	Broadcast	100%	75.24%
	Cross	100%	54.43%

Table 2 shows the results of using the strategy of Test5 with different portal rules. From the results we can see that this best damage-derived function can also achieve good results for different portal rules, thus vindicating the TDBU model.

4.3.2 With Barrier and Pheromone Decay

To further test the robustness of the design we implemented the best strategy found (Test5) under different environment and portal rule conditions, by adding a barrier to the environment as shown in Figure 3, and allowing pheromone values to decay as discussed previously. The maximum pheromone value was set to $V_{max} = 8$, and the pheromone value decays 0.4 every 60 time steps. The results of all three simulations are shown in Table 3.

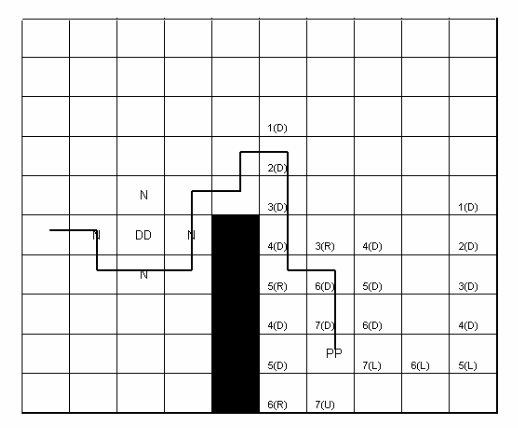


Fig. 3. The simulation environment with barrier

Table 3. Performance for different portal rules and environments

Environment	Portal Rules	S_R	E
Pheromone Decay, No Barrier	Ants	100%	40%
	Broadcast	100%	46%
	Cross	100%	41%
No Pheromone Decay, with Barrier	Ants	100%	39.69%
	Broadcast	100%	65.2%
	Cross	100%	43.7%
Pheromone Decay, with Barrier	Ants	95%	34%
	Broadcast	95.4%	35%
	Cross	95.6%	33.66%

As the results show, the original design is remarkably resilient, both to the addition of a barrier and to pheromone decay. Only when both properties are present does performance fall significantly, with success rates of less than 100%. Further work is necessary with a range of boundary configurations and pheromone decay rates, leading eventually to dynamic simulations when the new damage and portal shifts happen in real time.

5 Conclusions

We have demonstrated, for a simulated multi-agent sensor network, a design approach for the problem of robust and timely reporting of critical damage in the network, in a variety of environments including communications barriers and unknown (and time-varying) reporting sites (portals). A top-down/bottom-up (TDBU) approach, together with a genetic algorithm, has been successfully used to design properties of identical agents which are capable of reporting critical damage. From the simulations carried out we have seen that the design is far better than random searching method, and for the original design environment (no barriers, constant pheromones) gives average reporting times only twice that of the best possible when the portal location is known. We have also verified the robustness of the TDBU design for environments with barriers and with decaying pheromones. Remarkably, barriers or pheromone decay caused only a small decrease in reporting efficiency, and only when both were present did the robustness decrease significantly.

Further improvements can be expected with future research, for example by extending agent capabilities with more agent memory, defining new reporting functions with more parameters and adding pheromones to the report path. Significantly, thus far only the damage-derived agent parameters have been optimised. Joint optimisation of these and portal-derived parameters using co-evolution will maximise the usefulness of the TDBU approach.

References

1. Estrin, D., Govindan R., Heidemann J., Kumar S.: Next Century Challenges: Scalable Coordination in Sensor Networks. Proceedings of the Fifth Annual International Conference on Mobile Computing and Networks (MobiCOM '99), Seattle, August 1999.
2. Abbott, D., Doyle B., Dunlop J., Farmer T., Hedley M., Herrmann J., James G., Johnson M., Joshi B., Poulton G., Price D., Prokopenko M., Reda T., Rees D., Scott A., Valencia P., Ward D., and Winter J.: Development and Evaluation of Sensor Concepts for Ageless Aerospace Vehicles. Development of Concepts for an Intelligent Sensing System. NASA technical report NASA/CR-2002-211773, Langley Research Centre, Hampton, Virginia.
3. Vemuri, V.: Modeling of Complex Systems: an Introduction. New York (1978).
4. Mjolsness, E. and Tavormina A.: The Synergy of Biology, Intelligent Systems, and Space Exploration. IEEE Intelligent Systems - AI in Space, 3/4 2000.
5. Gadomski A. M., Balducelli C., Bologna S. and DiCostanzo G.: Integrated Parallel Bottom-up and Top-down Approach to the Development of Agent-based Intelligent DSSs for Emergency Management. The Fifth Annual Conference of The International Emergency Management Society, Washington, D.C. , May 19-22, 1998.
6. Guo, Y., Poulton P., Valencia P., and James, G.: Designing Self-Assembly for 2-Dimensional Building Blocks, ESOA'03, Melbourne, July 2003.
7. Poulton, G., Guo, Y., James, G., Valencia, P., Gerasimov, V., and Li, J.: Directed Self-Assembly of 2-Dimensional Mesoblocks using Top-down/Bottom-up Design. The Second International Workshop on Engineering Self-Organising Applications (ESOA'04), 20TH July 2004, New York, USA.

8. Kochhal M., Schwiebert L. and Gupta S.: Role-Based Hierarchical Self Organization for Wireless ad hoc Sensor Networks. Proceedings of the 2nd ACM International Conference on Wireless Sensor Networks and Applications, San Diego, CA, USA, 2003, pp: 98 – 107
9. Dorigo, M., Maniezzo V. and Colorni A.: The Ant System: An Autocatalytic Optimizing Process. Tech. Report No. 91-016 Revised, Politecnico di Milano, 1991.
10. Goldberg D.E.: Genetic Algorithms in Search, Optimisation, and Machine Learning. Addison-Wesley Publishing Company, Inc., Reading, Massachusetts, 1989.
11. Garis H.: Artificial Embryology: The Genetic Programming of an Artificial Embryo, Chapter 14 in book "Dynamic, Genetic, and Chaotic Programming", ed. Soucek B. and the IRIS Group, WILEY, 1992.