Available online at www.sciencedirect.com
SCIENCE DIRECT®
Robotics and Autonomous Systems 53 (2005) 36–58
Robotics and Autonomous Systems
www.elsevier.com/locate/robot

On connectivity of reconfigurable impact networks in ageless aerospace vehicles

Mikhail Prokopenko*, Peter Wang, Mark Foreman, Philip Valencia, Don Price, Geoff Poulton

Information and Communication Technologies Centre, Industrial Physics, Commonwealth Scientific and Industrial Research Organisation (CSIRO), Locked Bag 17, North Ryde 1670, Australia

Received 18 February 2005; received in revised form 7 March 2005; accepted 6 June 2005
Available online 8 August 2005

Abstract

The research results presented in this paper were obtained as part of the joint CSIRO-NASA Ageless Aerospace Vehicle (AAV) project. We describe the underlying principles, methodology, and preliminary results of modelling and simulating a multi-cellular sensor and communication network in a dynamic decentralised setting, motivated by a self-monitoring, self-repairing AAV. Such networks are expected to detect and react to multiple impacts and damage over a wide range of impact energies. In particular, we address the problem of forming a reconfigurable network (a minimum spanning tree) connecting cells that detected non-critical impacts, in presence of connectivity disruptions caused by critical impacts. The presented algorithm is based on the ant colony metaphor and may be complemented by gradient-based techniques. In addition, we measure the robustness of *impact networks* and present quantitative metrics that clearly identify phase transitions in network connectivity, separating chaotic dynamics from ordered and robust patterns.
© 2005 Elsevier B.V. All rights reserved.

Keywords: Self-organisation; Multi-agent networks; Impact detection; Minimum spanning tree; Ant Colony Optimisation; Phase transition

1. Introduction and background

Engineering structures of the future, whether they are vehicles, buildings, infrastructure, networks, etc., or heterogeneous groups of such structures, will be required to sense damage as it occurs, to evaluate the nature and severity of damage, to infer its cause (diagnosis), and to make a prediction (prognosis) of damage development and its effects. They will have the capability to make autonomous decisions for remedial actions, ultimately including self-repair. This paper investigates an approach to self-monitoring, self-repairing aerospace structures, motivated by NASA's vision of *ageless aerospace vehicles*—adaptive

* Corresponding author. Fax: +61 2 9325 3200.
E-mail addresses: mikhail.prokopenko@csiro.au (M. Prokopenko), peter.wang@csiro.au (P. Wang), mark.foreman@csiro.au (M. Foreman), PhilipValencia@csiro.au (P. Valencia), don.price@csiro.au (D. Price), geoff.poulton@csiro.au (G. Poulton).

0921-8890/$ – see front matter © 2005 Elsevier B.V. All rights reserved.
doi:10.1016/j.robot.2005.06.003

systems with a high degree of robustness and scalability [1–3,31]. In order to address these requirements we have chosen to distribute sensing, actuation and computational capabilities throughout the structure to form a complex multi-agent network, enabling the desired responses to self-organise within the complex system, with no central control.

The particular multi-agent system considered in this paper is a group of contiguous agents (referred to as "cells"), locally connected and forming the surface of a three-dimensional object. The cells not only form a physical shell for a structure, but also have in-built sensing and computational capabilities. Importantly, each cell can communicate only with its immediate neighbours, i.e., all communications, both local and global, need to occur through these cell-to-cell links. Although such constraints impede the flow of information, there is a significant potential redundancy which can aid robustness.

In this paper we describe the underlying principles, methodology, and preliminary results of modelling, simulating and developing a multi-agent Ageless Aerospace Vehicle (AAV) Concept Demonstrator, where cells self-organise into new groups (e.g., impact networks) in response to new information obtained from their sensors and communication ports. The *emergence of global response* as a result of interactions involving transfer of information embedded locally helps to completely avoid or reduce the number of single points-of-failure. In addition, we propose that the behaviour of each cell should be as simple as possible, in terms of both the internal logic and the communication policies. This principle—*economy of information*—is directly related to the ability to manufacture new cells and repair or replace damaged cells: the simpler the cell, the easier it is to repair/replace it.

Emergence is an expected (but not guaranteed) property of self-organisation, while the latter is typically defined as the evolution of a system into an organised form in the absence of external pressures. For example, Bonabeau et al. [5, p. 189]described self-organisation as

a set of dynamical mechanisms whereby structures appear at the global level of a system from interactions among its lower-level components. The rules specifying the interactions among the system's constituent units are executed on the basis of purely local information, without reference to the global pattern, which is an emergent property of the system rather than a property imposed upon the system by an external ordering influence.

One classical example is the formation of diverse spatial structures by groups of ants, described by Deneubourg and Goss [11], and the growth and morphogenesis of networks of galleries in the ant Messor sancta, resulting from a self-organised process that also allows the collective adaptation of the size and shape of the network to the size of the colony [7]. Several control algorithms for metamorphic robotic systems capable of spatial reconfiguration, exemplifying the use of emergent behaviour, are described by Bojinov et al. [4]: the multi-agent control achieves suitable reconfiguration as a "side-effect" of creating a structure with the properties (structural, morphological, etc.) required for a global task. Their agents (modules) have limited computational capabilities, communicating only with immediate neighbours, and using local rules to produce adequate control algorithms. A quite sophisticated multi-layered hierarchical motion planning strategy for a class of self-reconfigurable modular robotic systems is also proposed by Prevas et al. [28]. A novel class of locally connected mobile automata networks capable of a propulsive motion of the network's elements is described by Wessnitzer et al. [38]—a collective of mobile agents is shown to dynamically organise into simple spatial structures while the network evolves toward task-specific topologies.

Recent advances in sensor networks and micro-electro-mechanical devices led to the idea of localised algorithms, in which simple local node behaviours achieve a desired global objective [19], while communicating only with nodes within some neighbourhood. Estrin et al. proposed a simple communication model for describing localised algorithms—*directed diffusion* [19]. According to this approach, network nodes propagate *interests* that establish gradients directing the diffusion of data. As it propagates, data may be locally cached and/or transformed at each node. However, despite some progress, there is a lack of a unifying methodology underlying design of localised algorithms. The main question is how to produce and retain desirable emergent behaviour while avoiding potentially damaging patterns of agents' interaction. The problem of systematically transforming a global task

to individual behaviour models is yet to be resolved, although some promising results have been reported by Nagpal [27], in context of amorphous computing, where programmable self-assembly was demonstrated using biologically inspired multi-agent control. The contribution of Nagpal's work is in defining a small class of primitives (e.g., gradients, neighbourhood query, polarity inversion, cell-to-cell contact, etc.), a set of global operations, and a translation implementing the global operations as localised algorithms using the set of primitives. However, the question of how to obtain and inter-connect global operations themselves is left unanswered. Moreover, the task of discovering the correct primitives and translating them into localised agent programs would have to be repeated for each domain, in the absence of generic techniques and guidelines. A more general approach is described by Klyubin et al. [24], suggesting that selection pressures towards adaptability and robustness are related to (maximisation of) the information transfer within the agents' perception-action loops, enhancing the acquisition of information from (and through) the environment.

Thus, the problem of global response engineering in multi-agent networks is of central importance to this project, and motivates our search for generic selection pressures shaping self-organisation. Our proposed methodology is based on an iterative process that includes the following steps:

(a) forward simulation (for a class of localised algorithms dealing with impacts of various strengths), leading to emergent behaviour;
(b) quantitative measurement of emergent behaviour, based on identifiable phase transitions;
(c) evolutionary modelling of the desired global emergent behaviour, where the fitness functions correspond to the metrics obtained at step (b).

This paper models the steps (a) and (b).

The following section introduces a specific problem investigated in this paper: emergence of a rectilinear minimum spanning tree (RMST) as an impact network connecting cells that detected non-critical impacts. One localised algorithm dealing with this problem belongs to the class of Ant Colony Optimisation (ACO) algorithms (described in Section 3), and is presented in Section 4. Experimental evaluation of the algorithm is based on phase transitions in the problem's graph connectivity, and is described in Section 6. An alternative algorithm is chosen from the class of distributed dynamic programming algorithms and is contrasted with the ACO-based algorithm in terms of communication overhead and quality of produced RMSTs (Section 6.2), followed by some concluding remarks in Section 7.

2. Decentralised detection of multiple impacts

2.1. Architecture of the AAV concept demonstrator

In studying conditions leading to self-organising impact networks, we followed an incremental approach, trying to impose as few requirements as possible. A two-dimensional array of cells is modelled in a Concept Demonstrator (CD) [2,3]. We also used a stand-alone Asynchronous Simulator capable of simulating some environmental effects such as particle impacts of various energies. Fig. 1 shows the fundamental components of an autonomous agent (or cell), including four piezoelectric sensors and four communication ports.

The physical CD structure is a hexagonal prism formed from a modular aluminium frame covered by 220 mm × 200 mm, 1 mm thick aluminium panels that form the outer skin of the structure. Each such panel contains four "cells", and each of the six sides of the prism contains eight of these panels (Fig. 2). The skin therefore consists of 48 aluminium panels and 192 cells. Cells are the fundamental building blocks of the system, containing the sensing, processing and communication electronics. Each cell occupies an area of 100 mm × 100 mm of the skin, mounted on the inside of which are four piezo-electric polymer (PVDF) sensors. The sensors detect the acoustic waves that propagate through the skin as a result of an impact [3].

The AAV Asynchronous Simulator models each cell as an independent thread with a dedicated slice of processing resources. Cells are represented as objects (squares) on a two-dimensional plane, where they interact only with their immediate neighbours through connected (geometrically overlapping) communication ports. The decision on the next state of each element is a result of the interactions with immediate neighbours, and depends upon a localised algorithm embedded in the cell's logic.

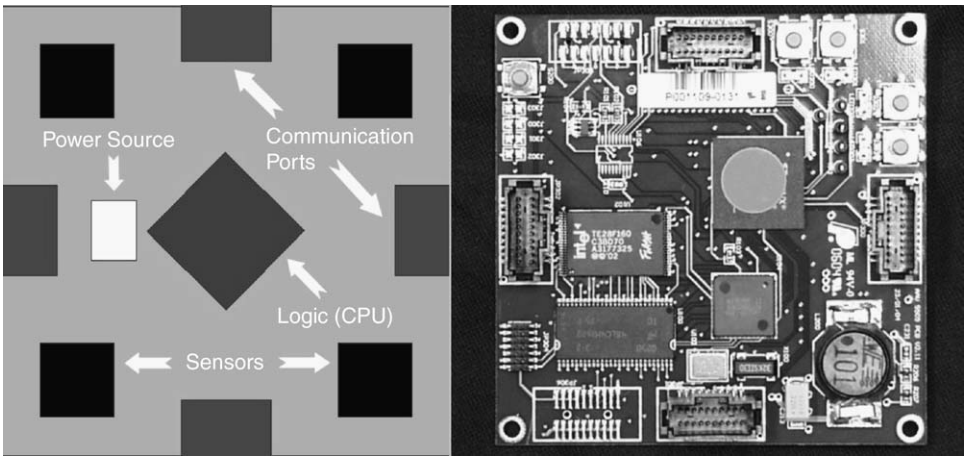

Fig. 1. A single square cell with four communication ports. (Left) AAV Simulator view. (Right) A CD view of a network application sub-module (four serial ports connecting to the neighbouring cells can be seen).

A biologically inspired impact detection mechanism embedded in a modular multi-cellular AAV skin enables the skin to determine impact boundaries and spread [26,21,35] in the presence of connectivity disruptions and cell failures—analogous to the clotting of a wound on a mammal and the regeneration of neurons by re-growing severed axons within a myelin sheath (made by Schwann cells). This work analyses dynamic formation of reconfigurable and robust *impact networks* connecting remote AAV cells that belong to a specific class. For example, impact networks may pre-optimise decentralised inspection by connecting cells that registered impacts within a certain band (e.g., non-critical impacts). When dealing with non-critical impacts, there is a possibility to extract information directly from the cells that register the impact—i.e., by using the impacted cells as network nodes. The inspection agents (communication packets or, potentially, swarming robots) may use an impact network in order to quickly explore the area and evaluate the damage (e.g., identify densities of impacts typical for a meteor shower, evaluate progression of corrosion, or to trace propagation of cracks). This is particularly important when a number of individually non-critical damage sites may collectively lead to a more serious problem. Another sub-task (e.g., decentralised repair) may need a self-organising network connecting local hierarchs responsible for coordination and routing of repair resources. An impact network should reconfigure around areas which may be inaccessible or damaged by

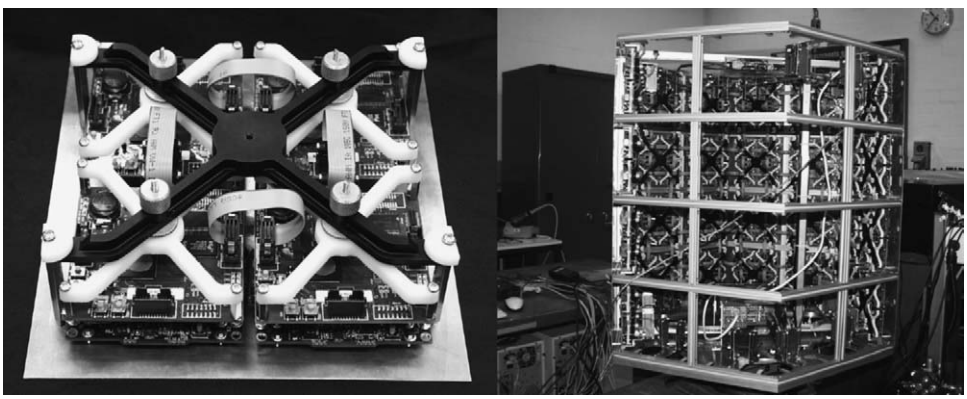

Fig. 2. (Left) Aluminium panel containing four cells. (Right) The CD with half of the panels removed to reveal the internal structure.

(sub-)critical impacts, destroying cells close to the point of the impact and affecting other cells. At the same time, the network should be sufficiently robust in order to support uninterrupted navigation of inspection agents or routing of repair resources.

2.2. Graph-theoretic problem representation

Robotic agents may need an impact network which solves a travelling salesperson problem (TSP). On the other hand, a shortest or *minimum spanning tree* is often required in order to enable decentralised inspections when virtual (software) agents are employed.

Let us define an AAV impact network. A two-dimensional AAV array can be represented by a planar grid graph $G(V, E)$: the product of path graphs on m and n vertices, which are points on the planar integer lattice, connected by the edges $E(G)$ at unit distances (Fig. 3). The cells which represent specific points of interest (e.g., the cells which detected non-critical impacts) form a subset P of $V(G)$. In order to produce an impact network, we need to identify those edges Z in $E(G)$ which connect the vertices in P minimally, so that the total distance (a sum of unit distances assigned to edges Z) is shortest. This problem is, essentially, the standard minimum spanning tree (MST) problem, except that a spanning tree is defined for a graph, and not for a set of vertices. Our problem is sometimes referred to in literature as the rectilinear minimum (terminal-) spanning tree (RMST) problem, while the vertices in P are called terminals, and is a fundamental problem in VLSI design [23]. The important difference between MST and RMST is that rather than choosing MST edges out of the graph edges $E(G)$ directly connecting pairs of vertices, we need to find multi-edge rectilinear paths between vertices in P, minimising the total distance. This can be achieved by means of an auxiliary complete graph A, whose vertex set is P and in which the edge pq for $p, q \in P$ with $p \neq q$ has length equal to the Manhattan distance between nodes p and q. The graph A is not a grid graph—it is an abstraction useful only to formalise an impact network: after a standard MST A_t is identified in A, we merely need to convert all edges in A_t to rectilinear paths on the grid graph G (Fig. 3).

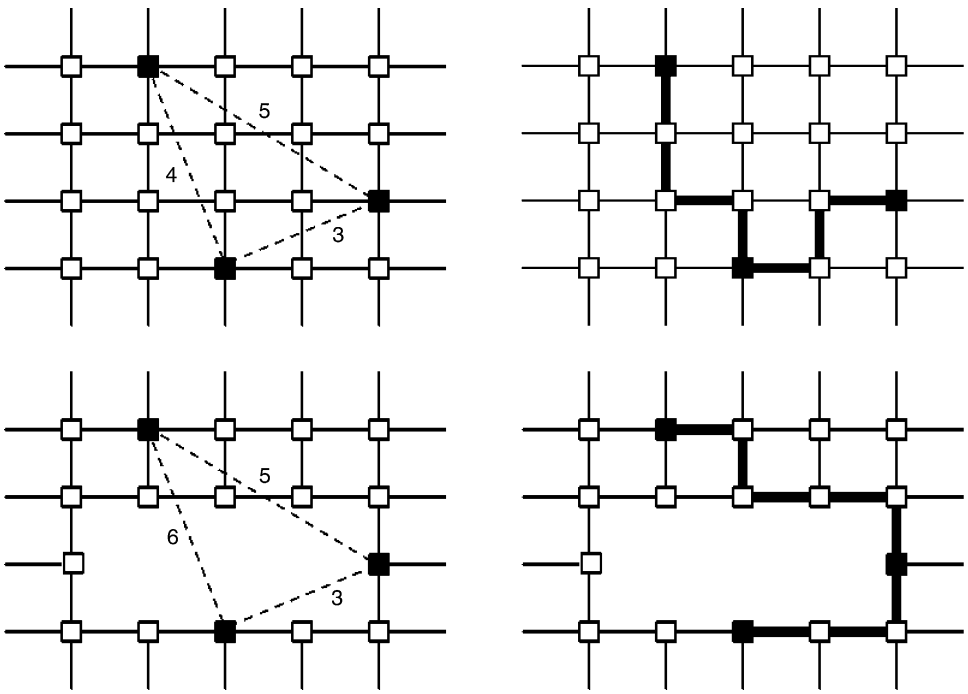

Fig. 3. Three impact nodes are shown in black. (Top-left) A complete auxiliary graph A with three edges shown as dashed lines. (Top-right) Conversion of its MST to rectilinear paths, shown as bold edges, in the AAV grid graph. Two lower figures show the AAV graph with some vertices removed. (Bottom-left) An updated auxiliary graph A. (Bottom-right) Conversion of the new MST to rectilinear AAV paths.

The impact network problem, however, is complicated by possible "obstacles", created by discontinuities in the AAV grid graph G. Initially, the grid graph G is solid—it does not have any "holes", so its complement in the infinite orthogonal planar grid is connected. New critical impacts may create such holes in the grid. Fig. 3 illustrates the RMST problem with two scenarios. The first case is shown in the top part, and involves three edges and a simple MST A_t with the total distance of 7. The second case is shown in the bottom part: some cells are destroyed (the corresponding vertices are removed), and the auxiliary complete graph should be updated because one shortest path has changed (from 4 to 6). This requires a recomputation of its MST (the new MST distance is 8), with another edge being selected and converted to a rectilinear path. This illustrates that a new obstacle may not just require that a new shortest path is found between the two involved cells (the problem investigated by Wu et al. [41]), but rather than the whole MST is re-evaluated. Moreover, there are cases when a cell/terminal is no longer needed to be included in the RMST, or a new cell/terminal needs to be added.

Thus, from a graph-theoretic standpoint, the representation of the impact network problem changes over time due to insertion of new nodes (e.g., non-critical impacts) or deletion of old nodes no longer fitting the impact range, while the problem's properties change due to varying connection costs (e.g., critical impacts destroying existing paths). In short, we need a dynamic and decentralised computation of a rectilinear minimum terminal-spanning tree in the presence of obstacles. If the information (such as the auxiliary graph A) was available at one central point, then the RMST problem would essentially become an MST problem, with a subsequent conversion to rectilinear paths. In this case the required computation itself would not be NP-hard, although the fully dynamic case, in which both insertions and deletions must be handled online, without knowing the sequence of events in advance, would still be quite hard. Eppstein [20] estimated a running time of a fully dynamic graph minimum spanning tree algorithm as $O(n^{1/2} + \log^2 n + n^{\epsilon})$, where ϵ is a (very small) constant, per update. In our case, the auxiliary graph A is not even known at any single node/cell. So, on the one hand, the desired algorithm should be both decentralised and fully dynamic, while on the other hand, our main concern is the communication cost of determining the desired topology.

In summary, this paper will focus on

emergence of rectilinear minimum spanning trees connecting AAV cells that detected non-critical impacts, while the decentralised problem's graph representation may change over time due to new non-critical impacts, and the problem's properties may change due to critical impacts destroying existing paths and causing connection costs to vary.

A localised algorithm solving this problem would exemplify a broader class of algorithms producing minimum spanning trees that connect cells sharing specific properties. Our ultimate goal is to develop and verify the methodology for evolving such algorithms—in other words, we should investigate not only algorithmic details, but also evaluation metrics identifying distinct phases in algorithm performance. These metrics will serve as fitness functions guiding the design towards desired objectives.

3. Candidate algorithms

3.1. Gradient-based cost fields

There are a few approaches employing gradient fields in the context of packet routing and querying algorithms in sensor networks, based on distance-vectors and link-state. The work of Ye et al. [42] covers some background material, and introduces the cost field concept: at each node, the cost field is defined as the minimum cost from that node to the event node (called *sink*). The cost field is established in a single pass—by choosing a minimal value between the broadcasted cost and the currently held value. More precisely, let us consider two nodes A and B, with the cost of messaging between them pre-set as $c_{A,B}$. Assume that the node B currently holds value l_B, and receives a message from the node A with the cost l_A (this is the cost of reaching the sink from A; initially all costs to the sink are set to infinity). The new value of l_B is then set to $\min(l_B, l_A + c_{A,B})$. This algorithm (GRADient broadcast) essentially forms gradients toward the event (sink). In order to reduce the flooding overhead of the required broadcast, the approach includes a timing mechanism to control the propagation. GRADient

broadcast attempts to capitalise on the similarity between the cost field and the natural gravity field "that drives water flowing from a high post to a low post" [42]. The crucial difference, however, is that, unlike gravity, the cost field is specific to each sink—there is no superposition among the fields established by different sinks. This feature, of course, makes the cost field concept less practical—especially, with multiple sinks: each node would have to keep (and propagate) costs to each sink. Besides, the static costs $c_{A,B}$ need to be maintained for each pair of nodes.

Another approach (rumor routing) described by Braginsky and Estrin [6], avoids establishing the cost field, and tries to balance propagation of queries and events. It extends the content of each packet with a table containing distances and directions to multiple events. At each newly reached node this table is synchronised (merged) with a local table, and the merged table is propagated further. Although the rumor routing algorithm is more efficient than some link-state algorithms that essentially require a distributed and replicated database dynamically mapping the whole network, it still creates and maintains multiple fields (as does GRADient broadcast)—unlike the single natural gravity field.

While elements of these algorithms can be useful in determining the complete (or partial) auxiliary graph A, they do not explicitly target a specific topological outcome like an RMST. The algorithms which do solve RMST and even harder problems, mostly work with full static and/or centralised knowledge of the graph (network), without taking into the account communication costs of discovering the required information. In either case, the outcome becomes available only in a single place rather than within the network itself.

3.2. Ant Colony Optimisation

It is now well known that the Ant Colony Optimisation (ACO) algorithms proposed and enhanced over the recent years by Dorigo and his colleagues [13,8,14,17] are particularly well suited for the following types of shortest path problems [17]:

- NP-hard problems, for which the dimension of the full state-space graph is exponential in the dimension of the problem representation—e.g., optimisation problems such as the Travelling Salesman Problem (TSP) [15] and Minimum Power Broadcast (MPB) [10];
- those *dynamic* shortest path problems in which the properties of the problem's graph representation change over time concurrently with the optimisation process, that has to adapt to the problem's dynamics;
- those *decentralised* problems in which the computational architecture is spatially distributed.

Although the problem of forming minimum spanning trees (impact networks) on the AAV skin is not NP-hard, it is spatially distributed over a grid comprising a significant number of cells. Even more importantly, as mentioned earlier, the nodes that need to be included in the minimum spanning tree (non-critical impacts) may appear at any time, changing the graph representation. Concurrently, critical impacts may destroy some existing paths and force the network to reconfigure around new obstacles, thus changing connection costs. In short, the process should adapt to the problem's dynamics in a decentralised setting.

These factors suggested that the problem of forming minimum spanning trees on the AAV skin can be efficiently tackled by ACO algorithms, rather than distributed dynamic programming (Bellman-Ford) algorithms or reinforcement learning techniques such as backtracking. A comparison with a distributed version of Prim's MST algorithm is described in Section 6.2. Overviews of the ACO meta-heuristic and its applicability can be found in [17,9]. Essentially, the ACO algorithms use the ability of agents to indirectly interact through changes in their environment (stigmergy) by depositing pheromones and forming a pheromone trail. They also employ a form of *autocatalytic* behaviour—*allelomimesis*: the probability with which an ant chooses a trail increases with the number of ants that chose the same path in the past. The process is thus characterised by a positive feedback loop [13,14]. The most interesting aspect of this autocatalytic process in the context of the AAV is that finding the shortest path around an obstacle is an emergent property of the interaction between the obstacle shape and the ants distributed behaviour.

Each cell/node in the network stores an ant-routing table, used in determining which neighbour cell should be chosen by an incoming ant packet to continue their travel. The ant-routing table $a_{i,j}(t)$ of node it i with respect to all its neighbour nodes j, contained in the

neighbourhood N_i, is usually obtained by the following functional composition of pheromone trails $\varphi_{i,j}(t)$ and local heuristic values $\eta_{i,j}(t)$:

$$a_{i,j}(t) = \frac{[\varphi_{i,j}(t)]^{\alpha}[\eta_{i,j}(t)]^{\beta}}{\sum_{l\in N_i}[\varphi_{i,l}(t)]^{\alpha}[\eta_{i,l}(t)]^{\beta}} \quad \forall\, j \in N_i,$$

where α and β are two parameters balancing relative weight of pheromone trail and heuristic value. When distances between neighbour nodes are not the same, the local heuristic $\eta_{i,j}(t)$ may reflect the *visibility*, determined as the inverse distance between nodes i and j, and indicating that closer nodes are more strongly preferred. It could also reflect other domain knowledge: for example, the length of the communication queue on the link connecting nodes i and j. The transition probability of the ant k from node i to node j at time t is given by the following probabilistic decision:

$$p^k_{i,j}(t) = \frac{a_{i,j}(t)}{\sum_{l\in N^k_i} a_{i,l}(t)},$$

where N^k_i is the feasible neighbourhood of node i for ant k, determined by the problem constraints.

The intensity of trail $\varphi_{i,j}(t)$ on the path between nodes i and j gives information on how many ants have chosen the path in the past. The trail intensity is updated each time an ant agent k passes through the node:

$$\varphi_{i,j}(t) = \varphi_{i,j}(t) + \Delta\varphi^k_{i,j}(t).$$

At the beginning (in some ACO variants, at the end) of each cycle, the pheromone evaporates at the rate $\rho \in (0, 1]$:

$$\varphi_{i,j}(t) = (1 - \rho)\varphi_{i,j}(t).$$

For simplicity, we shall use $\varrho = 1 - \rho$ to denote the pheromone retention coefficient: the higher ϱ corresponds to higher retention of pheromones. Sometimes, the possible values for the pheromone trails are limited to the range $[\varphi_{\min}, \varphi_{\max}]$. There are also different ways to choose quantity $\Delta\varphi^k_{i,j}(t)$, while updating trail intensity. According to the simplest scheme (ANT-density), this additional quantity is always constant, while the ANT-quantity scheme makes $\Delta\varphi^k_{i,j}(t)$ inversely proportional to the distance between nodes i and j, further reinforcing the visibility factor. Other variants (ANT-cycle) use, in addition, the information collected during past simulations to direct the search for better solutions. Algorithms in the Ant Colony Systems sub-class [16] balance exploration with exploitation by following sometimes a stronger local trail with probability of one, rather than using probabilistic decisions. These and many other recent ACO variants and trends are reviewed by Cordon et al. [9].

There is another approach inspired by the ant colony metaphor and the concept of stigmergy: AntNet—an approach to adaptive learning of routing tables in communication networks [12]. Stigmergetic communication, utilised by AntNet, allows a set of ant-like agents to implicitly coordinate the actions while asynchronously building probabilistic adaptive routing tables. AntNet has also been contrasted with both distributed dynamic programming and reinforcement learning algorithms. In particular, in AntNet, "following the generation of a stochastic transition chain by the forward ant there is no back-chaining of the information from one state ... to its predecessors. Each state is rewarded only on the basis of the ant's trip time information strictly relevant to it" [12]. Thus, this approach is completely different from temporal difference methods such as Q-Routing and its extensions. As noted by Di Caro and Dorigo [12], "these algorithms build the distance estimates at each node by using the predictions made at other nodes", and therefore, are not robust to wrong estimates—unlike AntNet which does not propagate local estimates to other nodes. In addition, probabilistic adaptive routing tables enable the ants to explore new, possibly better, paths. It was also experimentally observed that AntNet is more robust than its competitors to the frequency with which routing tables are updated [12]. In short, when the network properties change concurrently with the decentralised problem-solving process, the ant colony metaphor suggests a feasible solution.

The successes of ACO-based algorithms and AntNet provide a considerable motivation for our effort. Nevertheless, we need to account for specifics of our application: an emphasis on a particular topological outcome rather than routing from a source node to destination; a preference for availability of this outcome within the network itself rather than in a dedicated cell; highly dynamic impact scenarios with no a priori knowledge of the involved graph (e.g., edge costs); a

decentralised architecture without absolute coordinates or id's of individual cells on a multi-cellular aerospace vehicle skin, etc.

4. Self-organising impact networks: minimum spanning trees

In this section we describe our primary proposed solution. The ant-like agents follow an initial exploration phase and a subsequent dead-reckoning phase. The agents are implemented as communication packets themselves, so the policies are implemented via appropriate message passing, where the cells are responsible for unpacking the packets, interpreting them, and sending updated packets further if necessary. Thus, ants cannot move into the cells with damaged (or shutdown) communication links—therefore, the regions affected by critical impacts form obstacles, and the ants are supposed to find shortest paths around them using positively reinforced pheromone trails. Ants are usually generated with certain frequency, or at a given interval T—a given number of time steps (cycles), and terminate after a fixed number of cycles greater than T.

The exploration phase includes the following behaviours:

(E1) each impact node generates a number of exploring ants every T cycles; each ant has a "time to live" counter (e.g., 255), decremented every cycle;
(E2) at each step from cell i to cell j, an exploring ant updates the x- and y-shift coordinates from the home node (initially set to 0)[1];
(E3) an exploring ant performs random walk until either (a) another impact node is found, or (b) the ant has returned to the home impact node; or (c) the ant can move to a cell with a non-zero trail intensity;
(E4) if an exploring ant can move to a cell with a non-zero trail intensity, the destination cell is selected according to the transition probabilities; an exception being the very first move after joining a trail, when the ant simply prefers to start moving in the direction further away from home.

While performing random walk (E3), exploring ants exclude the cell where they were at the previous cycle, unless this is the only choice. Similarly, in calculating the transition probabilities used on the "follow-trail" step (E4), the cell from where the ant came is excluded from the feasible neighbourhood.

The investigated method also involves a dead-reckoning scheme (DRS), inspired by Tunisian desert ants studied by Wehner and Srinivasan [37], enabling the agents to return home when another impact node is located. The scheme is implemented as follows:

(R1) when another impact node is found, the exploring ant switches to a return state, remembers the ratio $g = y/x$ corresponding to the found node's coordinates relative to the home node, and starts moving back to the home node—moving to cells where the y- and/or x-shift coordinates(s) would be smaller and their ratio would be as close as possible to g; if both x- and y-shift are zero (the home node), the returning ant stops;
(R2) if the cell suggested by the dead-reckoning scheme (minimisation of x- and/or y-shift, while maintaining g) cannot be reached because of a communication failure (an obstacle),7/15/05 the ant selects an obstacle avoiding move according to the transition probabilities; upon this selection the ant keeps to the chosen path until the obstacle is avoided, as recognised by comparison of current y/x ratio with g;
(R3) each cycle, a returning ant deposits pheromone in the quantity inversely proportional to the traversed return distance d (d is incremented by 1 each cycle); the deposited pheromone is limited by a pre-defined maximum $\varphi_{\max}$; at the beginning of each cycle pheromone evaporates in each cell.

Basically, the rule (R3) reflects the *aging* strategy [33] which, together with *evaporation* strategy [17], reduces the effect of past experience and avoids convergence to possibly stagnant sub-optimal paths. In general, the ants going along a shorter return path deposit more pheromone than the ants that selected

[1] These coordinates are relative, they simply reflect how many cells separate the ant from the home node in terms of x and y at the moment, and should not be confused with a taboo list of an ACO agent containing all visited nodes in terms of some absolute coordinate or identification system.

(due to transition probabilities) a longer path around an obstacle—simply because the deposited quantity is inversely proportional to the traversed distance. A higher quantity of pheromone attracts more ants. Eventually, a trail is established between a pair of impact nodes. It is important to realise that in any such pair, both impact nodes generate exploring ants that potentially find the other impact node and return back. So the shortest trail is reinforced by returning ants going in opposite directions (Fig. 4).

The pheromone is deposited in the cells themselves rather than in communication links—in other words, we deal with pheromone trail intensities φ_j at the cell j. While calculating the ant-routing tables, we set $\alpha = 1$ and $\beta = 0$, resulting in

$$a_{i,j} = \frac{\varphi_j}{\sum_{l \in N_i} \varphi_l} \quad \forall\, j \in N_i.$$

The visibility factor is discarded because the DRS mostly subsumes its role. Obviously, in order to update its ant-routing table the cell i needs to receive individual pheromone-updates from its neighbours. These updates are delivered by cyclic communication messages and can be seen as *bounded pheromone dissipation*: the updates do not propagate.

To avoid retracing of its steps, an ant k located at the cell i remembers the node m^k where it was at the previous cycle, and excludes it from the feasible neighbourhood $N_i^k = N_i \setminus \{m^k\}$, unless it is the only choice. The transition probabilities used on the "follow-trail" step (E4) are calculated as follows:

$$p_{i,j}^k = \begin{cases} \dfrac{a_{i,j}}{\sum_{l \in N_i^k} a_{i,l}} & \text{if } \varphi_{m^k} \neq 0; \\ 1 & \text{if } \varphi_{m^k} = 0 \text{ and } |x_i| + |y_i| < |x_j| + |y_j|; \\ 0 & \text{if } \varphi_{m^k} = 0 \text{ and } |x_i| + |y_i| > |x_j| + |y_j|, \end{cases}$$

where the last two rules represent the exception for the very first move after joining a trail, driving the ant further away from home. The values x_i and y_i are the relative x- and y-shifts maintained by the exploring ant at every step, while x_j and y_j can be always determined locally, given a direction to the neighbour j relative to the cell i. This simple local heuristic does not always guide the ant away from home – particularly, when joining a trail behind an obstacle – but in general helps to send more ants towards their targets.

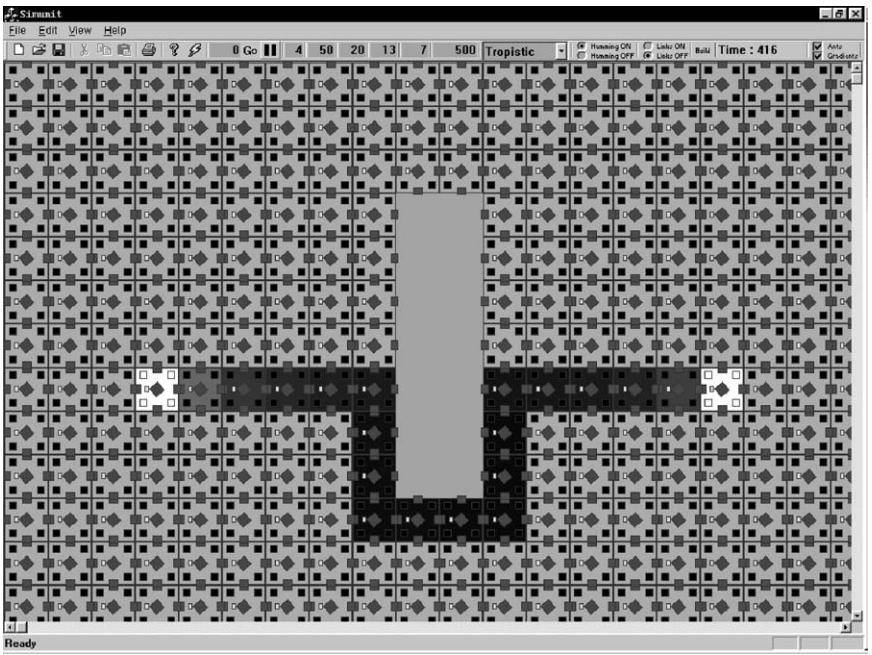

Fig. 4. An example of a shortest path between two impact nodes (white cells) around an obstacle – different shades of green (dark) colour represent the trail intensity – the lighter, the more intense.

The returning ants follow dead-reckoning scheme, and use transition probabilities only when they face a new obstacle, at the step (R2):

$$p_{i,j}^k = \begin{cases} \dfrac{a_{i,j}}{\sum_{l \in N_i^k} a_{i,l}} & \text{if } \exists l \in N_i^k,\ a_{i,l} > \overline{\varphi}; \\ \dfrac{1}{N} & \text{if } \forall\, l \in N_i^k,\ a_{i,l} \leq \overline{\varphi}, \end{cases}$$

where N is the number of nodes in the feasible neighbourhood N_i^k, and $\overline{\varphi}$ is a simple obstacle threshold encouraging more exploration in probabilistic decisions earlier in the process.

The investigated ACO-DRS algorithm, based on the rules (E1)–(E2)–(E3)–(E4)–(R1)–(R2)–(R3), approximates rectilinear minimum spanning trees (Figs. 5 and 6) in the presence of obstacles (the blocking problem). It also performs well in dealing with the shortcut problem [33], corresponding to a new shorter trail becoming available.

To reiterate, we implemented a modified ACO algorithm with

- one type of non-dissipative evaporating pheromone, limited by $\varphi_{\max}$;
- one type of aging ants with limited private memory, including variables like the relative x- and y-shift from the home node, the relative index m^k of the cell where the ant was at the previous cycle, and the ratio $g = y/x$ of the found node's coordinates relative to the home node;
- simple ant-routing tables containing normalised pheromone values only for immediate neighbours.

The modifications address the specifics of AAV and include the dead-reckoning scheme; the transition probabilities model with the obstacle threshold $\overline{\varphi}$, and the local heuristic used after joining a trail. A variant of the ACO-DRS algorithm, employing an adaptive dead-reckoning scheme (ADRS) and a novel local heuristic improving the algorithm convergence and performance, is described elsewhere [32].

5. Phase transitions in impact networks

In order to characterise dynamics of multi-cellular impact networks and convergence of our algorithm we

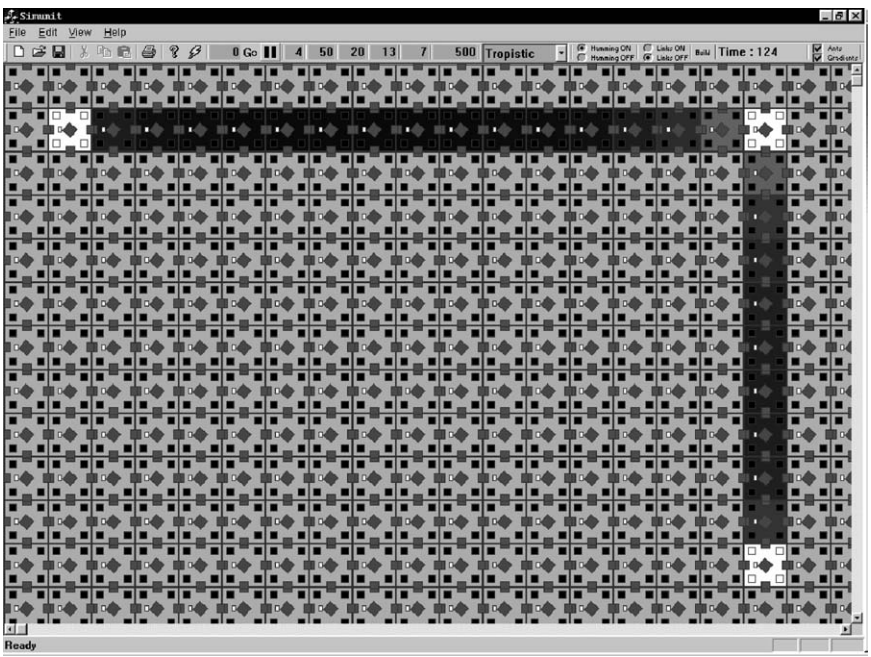

Fig. 5. An example of a minimum spanning tree in a simple impact network (impact nodes are white cells)—a "diagonal" trail is much weaker and its trail intensity is below a given threshold ϑ.

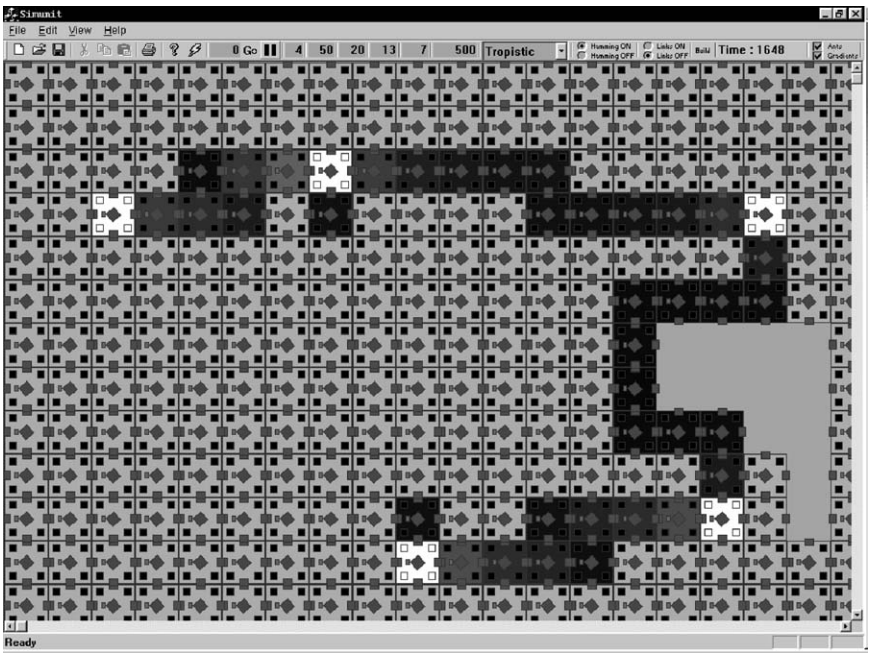

Fig. 6. An example of a reconfigurable impact network.

shall try to identify distinct phases and transitions between them—in terms of connectivity provided by the emergent spanning trees. It is well known that one feasible average measure of a complex network's heterogeneity is given by the entropy of a network defined through the link distribution:

$$H(P) = \sum_{k=1}^{m} P_k \log\left(\frac{1}{P_k}\right).$$

The latter can be defined via the simple degree distribution—the probability P_k of having a node with k links, or via the *remaining* degree distribution [34]. Similarly, one can capture the average uncertainty of the network as a whole, using the joint entropy

$$H(P, P') = \sum_{k=1}^{m}\sum_{k'=1}^{m} P_{k,k'} \log\left(\frac{1}{P_{k,k'}}\right),$$

where $P_{k,k'}$ is the joint probability of connected pairs. Ultimately, the amount of correlation between nodes in the graph can be calculated via the mutual information measure (the information transfer [34]) as

$$I(P) = H(P) - H(P|P') = \sum_{k=1}^{m}\sum_{k'=1}^{m} P_{k,k'} \log\frac{P_{k,k'}}{P_k P_{k'}}. \quad (1)$$

The random graphs theory that dates back to the seminal works of Erdös and Rényi [18] suggests to use the size of the largest maximally connected subgraph (LCS) and its variance or standard deviation obtained across an ensemble of graphs, as an indicator of phase transitions in the graph connectivity. It is well known that critical changes in connectivity of a directed graph occur as the number of edges increases—the size of the LCS rapidly increases as well and fills most of the graph, while the variance in the size of the LCS reaches a maximum at some critical point before decreasing [22]. This variance approximates mutual information contained in the dynamics, combining both individual cells' diversity and their interdependence (correlation). In general, the mutual information defined by (1) is a better, more generic measure of dependence:

correlation functions, like the variance in the size of the LCS, "measure linear relations whereas mutual information measures the general dependence and is thus a less biased statistic" [34]. However, for our purposes it is more practical and sufficient to characterise the diversity of and interdependence within impact networks through the variance in the size of the LCS. It could be argued that only complex dynamics exhibits high variance, and the peak of this variance would point to a phase transition in connectivity.

Information-theoretic measures of complexity were also effectively used to categorise and classify distinct emergent configurations and phase transitions in the context of Cellular Automata (CA) [39,25,40]. Wuensche [40] has used variance of input-entropy over time in classifying the rule-space of one-dimensional CAs into ordered, complex and chaotic cases, related to Wolframs's qualitative classes of CA behaviour [39]. Langton [25] has also observed that the individual cell entropy shows a discrete jump between low and high entropy values for a particular value of the λ parameter (the ratio of cells with a given property, for example, "live" cells). This evidence pointed to a first-order phase transition, similar to that observed between the solid and fluid phases of matter. The average mutual information, defined as a function of individual cell entropies, has a distinct peak for this λ, indicating a phase transition between "ordered" and "chaotic" CA dynamics.

6. Experimental results

6.1. Impact network stability

The analysis of spatial stability of impact networks is based on the concept of a connected trail-fragment (CTF). A CTF is a set F of cells with $\varphi \geq \vartheta$ (where ϑ is a given threshold), such that every cell in F is connected with at least one other cell in F, and there exist no cell outside F which is connected to at least one cell in F (an analogue of a maximally connected subgraph). Tracing the average size H of CTFs and its the variance σ^2 over time corresponds to considering an ensemble of impact networks generated with different probabilities of connected impact nodes.

We focussed on two critical parameters: pheromone retention rate ϱ and the ants generation interval T. To re-iterate, retention rate ϱ determines how much pheromone is left in the cell at the end of each cycle ($\varrho = 1.0$ means that there is no evaporation), and T sets the interval for generating new ants by every affected cell—more precisely, every cell generated four ants for each interval T. We carried out a series of experiments with two impacts, fixing one critical parameter and varying the other, and repeating each experiment three times. During each experiment, we calculated the average size of CTF's in impact networks, $H(\varrho)$ or $H(T)$, at each time-point, and its standard deviation, $\sigma(\varrho)$ or $\sigma(T)$, over time. The series was also repeated with three impacts.

Figs. 7–9 plot the average CTF size $H(T)$ for three intervals. A short interval ($T = 20$) corresponds to a high frequency of the ants generation process, and results in robust "ordered" trails (Fig. 7), where the size quickly stabilises. A medium-length interval ($T = 70$) results in highly unstable trails (Fig. 8), with the most "complex" dynamics: a trail frequently forms and breaks. The longer intervals (e.g., $T = 180$) generate too few ants to support a trail for any significant time period (Fig. 9), and the ensuing dynamics can be characterised as "chaotic". Similar dynamics is observed when varying pheromone retention rates, producing robust trails for higher rates $\varrho \geq 0.97$, unstable trails for the medium rates around $\varrho = 0.93$, and chaotic trails for the lower rates $\varrho < 0.9$, where pheromone evaporates too quickly.

A more detailed picture emerges from Figs. 10 and 11, plotting results of a series of experiments with (a) two impacts as shown in Fig. 4, and (b) three impacts as shown in Fig. 5. Both plots indicate a clear peak in the standard deviation: $\sigma(T)$ peaks at medium-length intervals, and $\sigma(\varrho)$ peaks at rates $\varrho = 0.94$ or $\varrho = 0.95$. Robust trails occur for low intervals and high retention rates, in particular, $T = 20$ and $\varrho = 0.98$. The medium-range values, where we observe "complex" unstable dynamics, highlight a phase transition in network connectivity, and clearly separate ordered and desirable robust scenarios from chaotic and under-performing cases.

6.2. A comparison with distributed prim's RMST algorithm

In this section we briefly describe an alternative way of dealing with the problem. Our aim is to compare the

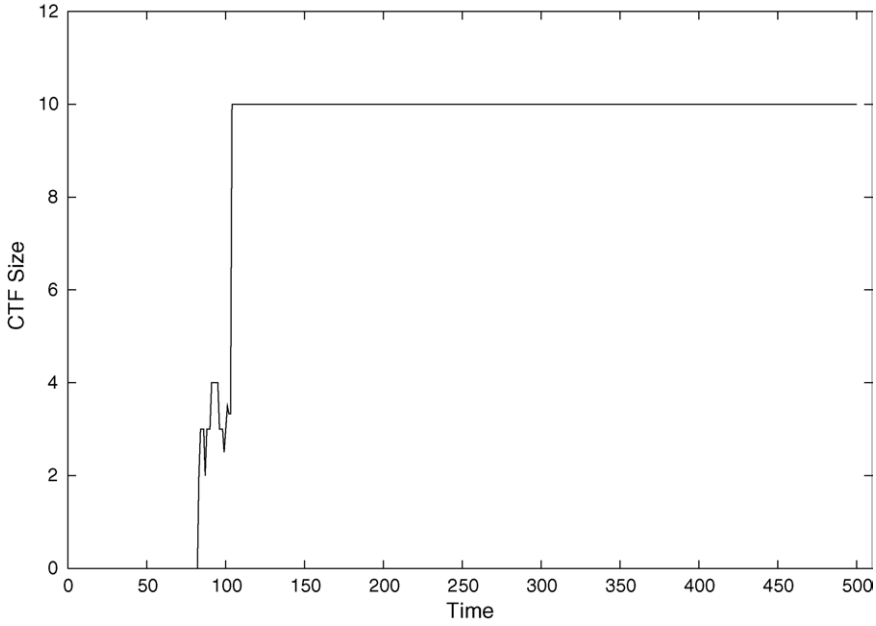

Fig. 7. Average size $H(T)$. Robust trails with $T = 20$.

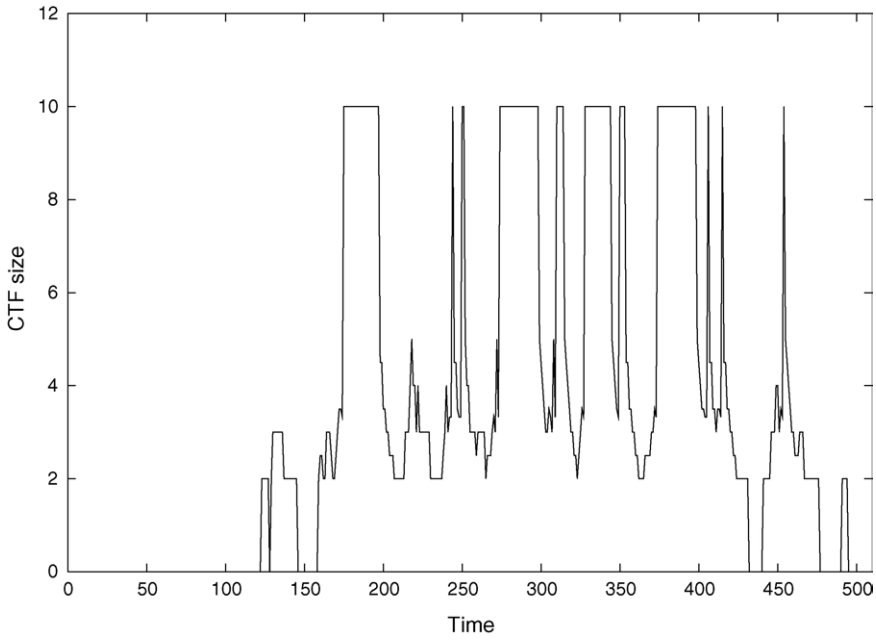

Fig. 8. Average size $H(T)$. Unstable trails with $T = 70$.

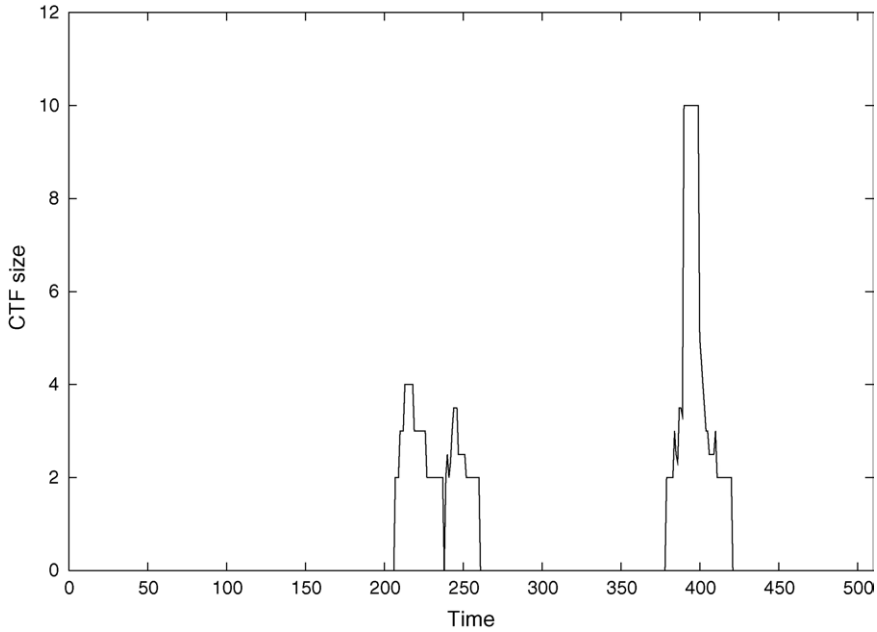

Fig. 9. Average size $H(T)$. Chaotic trails with $T = 180$.

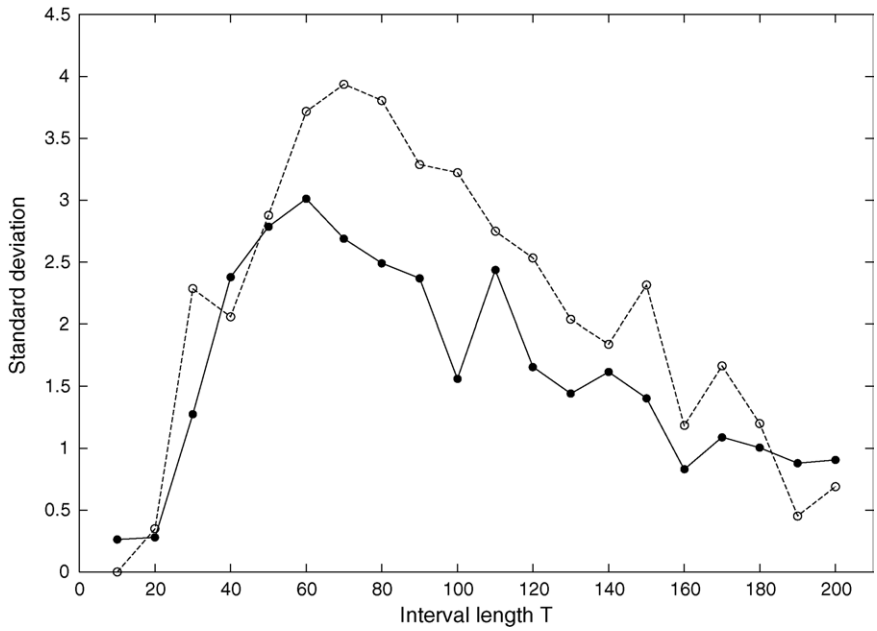

Fig. 10. Standard deviation $\sigma(T)$ of average size $H(T)$. The scenario with two impacts is indicated with dashed lines; three impacts: solid lines.

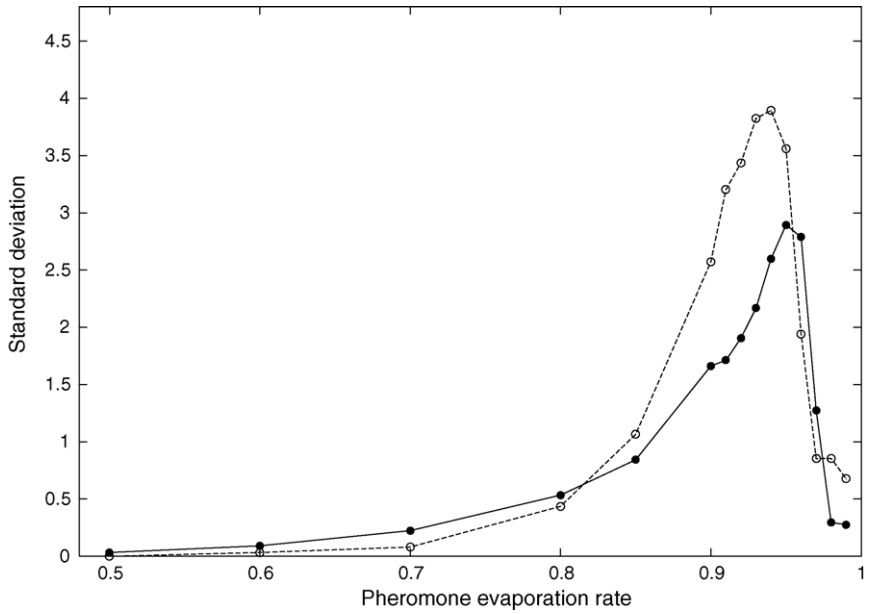

Fig. 11. Standard deviation $\sigma(\varrho)$ of average size $H(\varrho)$. The scenario with two impacts is indicated with dashed lines; three impacts: solid lines.

alternative with our ACO-DRS algorithm in terms of communication overhead and quality of MST approximation. The alternative approach is based on Prim's algorithm for computing an MST [29], embedded in a single dedicated cell. This cell plays a role of a global hierarch, and is obviously, a single point of failure. However, our comparative analysis does not penalise the centralised computation algorithm for this architectural weakness—instead we focus on the communication cost of discovering the required information.

In order to compute an MST, the global hierarch has to obtain information about the cells (vertices) P which need to be connected, and the paths (edges) among them. In other words, it has to process the auxiliary complete graph A, whose vertex set is P and in which the edge pq for $p, q \in P$ with $p \neq q$ has length equal to the Manhattan distance between nodes p and q. As we pointed out before, the lengths of the involved edges are not known a priori and have to be dynamically discovered before the RMST is computed. Many distributed dynamic programming algorithms calculate graph edges by setting gradients (cost-fields). Since our goal is to compare with an algorithm from this class, we obtain the complete graph A by setting gradients from each of the cells in P. Every such gradient is established in a single "flooding" pass initiated by an impact node $p \in P$ by broadcasting an integer, which is incremented at each message-passing step with an appropriate minimisation (based on the GRADient broadcast idea). Every intermediate cell, therefore, may be a point on many overlapping gradient fields, and has to keep a list of integer gradient values, pointing towards many "sinks" (vertices in P). When a gradient initiated by a node $p \in P$ reaches another node $q \in P$, the node q effectively determines the distance to p (the last gradient value), and may send an update to the global hierarch. This means that every node in P is expected to establish distances to all other nodes in P, and the global hierarch is expected to obtain the complete graph A. Given that the exchanged data become outdated when there is a change (a new node is added to P, an existing node is removed, an obstacle affects costs of edges), the complete graph A will have to be updated from time to time. To do so, every gradient is periodically re-established, distances between the affected nodes are modified, and the updates are communicated. Similarly, every node in P periodically sends data to the global hierarch just to confirm its membership (otherwise, it is removed from the task). At certain times, the global hierarch applies Prim's algorithm to the current graph A and computes the MST and RMST. A software inspection agent can then be send to explore the RMST.

Distributed periodic determination of the complete graph A incurs, obviously, the main communication cost of this algorithm. On the other hand, the quality of computed MST is expected to be higher than in the ACO-DRS algorithm. There would obviously be delays when an inspection agent traverses an RMST which is no longer correct—this can be resolved by more updates from the global hierarch sent directly to the inspection agent. There are also problems when the global hierarch is destroyed, and this can be resolved by redundancy in the hierarchy. All these resolutions increase the communication cost and algorithmic details, but at this stage we simply wished to compare the ACO-DRS algorithm with a "vanilla" version of distributed Prim's algorithm.

We carried out a number of evaluation experiments with two scenarios. The first scenario (S_1) involved a smaller AAV array (19×14 cells, 266 in total), while the other scenario (S_2) employed a larger array (30×20 cells, 600 in total). Each scenario simulated the following series of events (Fig. 12): (i) two impacts (A and B) are detected at the very beginning, and the expected RMST is a straight path between them: $\{(A, B)\}$; (ii) an obstacle appears some time later (at cycle 200 for S_1, and at 300 for S_2), and the expected RMST is the shorter path around it; (iii) a third impact (C) is detected 400 cycles after the obstacle, and the RMST is expected to include a path to the new impact: $\{(A, B), (A, C)\}$; (iv) a fourth impact (D) is detected 400 cycles after the third impact, and now the RMST should reconfigure to $\{(A, C), (C, D), (B, D)\}$, because the edge (C, D) is shorter than (A, B), given the obstacle; (v) the fourth impact (D) is ruled out 400 cycles later, and the RMST should reconfigure back to $\{(A, B), (A, C)\}$. The series of events (i)–(v) simulates all important aspects of the problem: addition and deletion of nodes, and changes in the edge costs. In the second scenario the ACO-DRS algorithm employed twice as many exploring ants per impact node.

Each scenario included 10 runs for the ACO-DRS algorithm and 1 run for the distributed Prim's algorithm, as the latter is essentially deterministic. Fig. 13 illustrates the comparative quality of the algorithms, and it is obvious that the alternative performs significantly better than the ACO-DRS algorithm: it converges to the right RMST length faster and does not fluctuate (it is particularly better when dealing with node deletion). This outcome is, of course, expected,

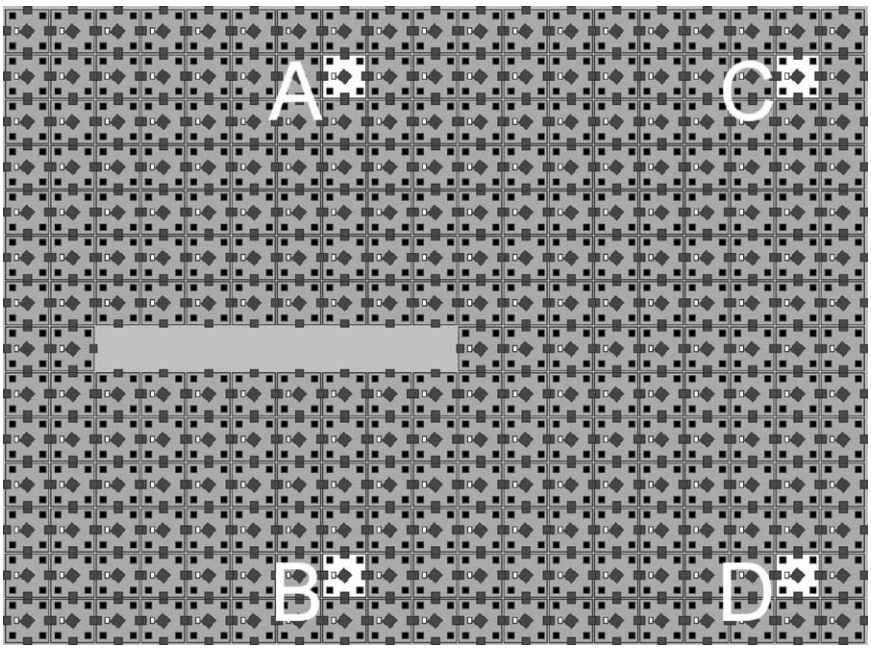

Fig. 12. The first scenario: (i) A and B are added; (ii) the obstacle is added; (iii) C is added; (iv) D is added; (v) D is removed.

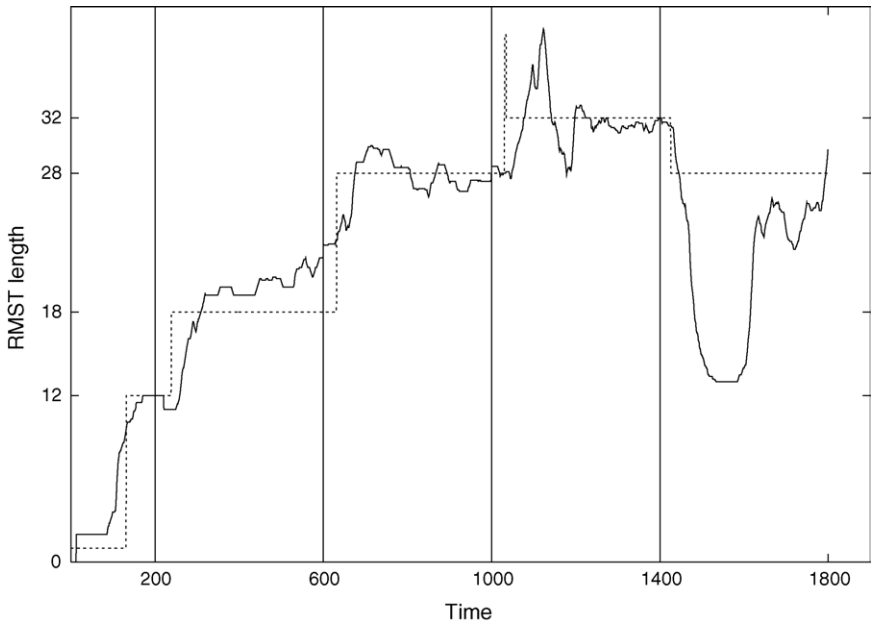

Fig. 13. Length of RMST produced by the algorithms for the first scenario: ACO-DRS length is shown with solid lines, global-hierarch computation is shown with dashed lines.

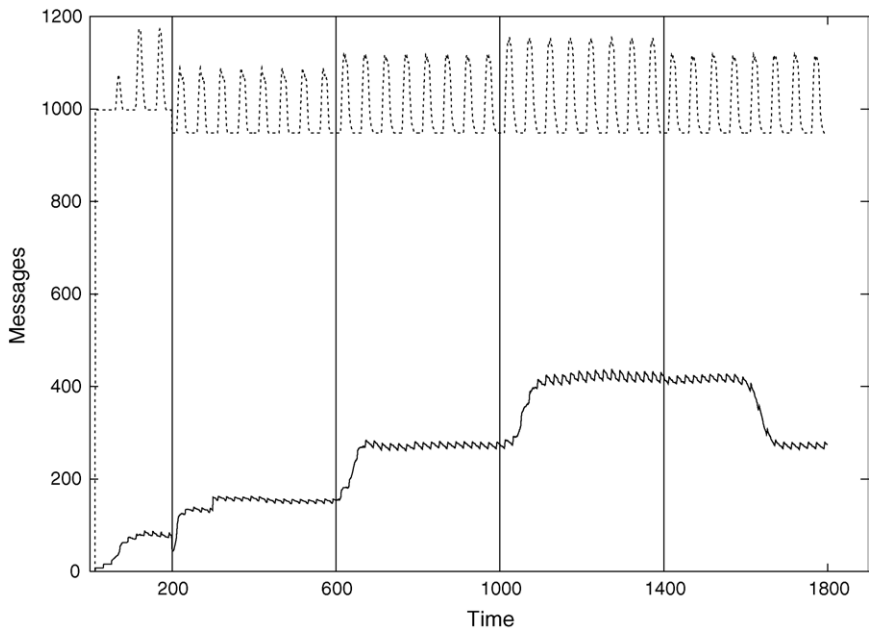

Fig. 14. Communication load created by the algorithms for the first scenario. ACO-DRS: solid lines; global-hierarch: dashed lines.

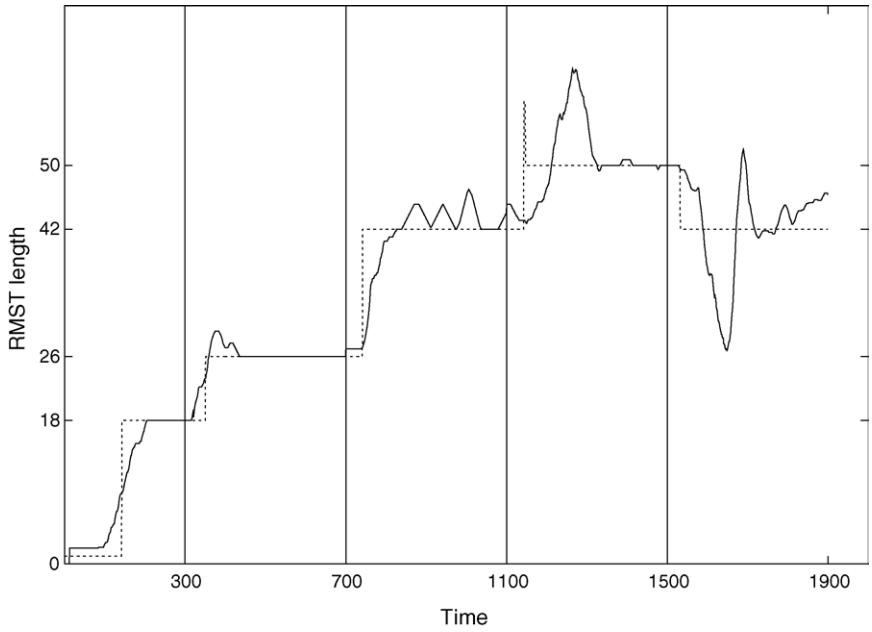

Fig. 15. Length of RMST produced by the algorithms for the second scenario. ACO-DRS: solid lines; global-hierarch: dashed lines.

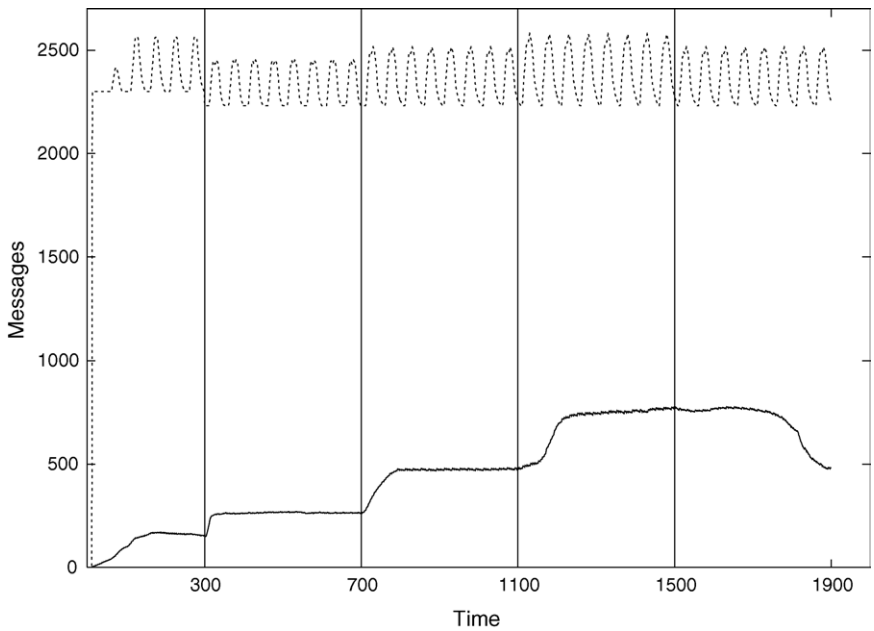

Fig. 16. Communication load created by the algorithms for the second scenario. ACO-DRS: solid lines; global-hierarch: dashed lines.

as the RMST computation is carried out in the global hierarch, and not in the network itself. Fig. 14 shows that the price for this performance is quite high: the distributed Prim's algorithm requires almost an order of magnitude more messages than the ACO-DRS algorithm. Similar picture emerges for the second scenario (Figs. 15 and 16). Here, however, the gap in the communication overhead widens, while the performance gap is reduced.

In order to verify whether these observations indicate a tendency, we need to find out the main component of the communication overhead in both algorithms. The ACO-DRS algorithm involves messages carrying exploring and dead-reckoning ants, and the messages updating neighbours of each cell on the value of its pheromone. It is this last message type which happens to be the most costly in the ACO-DRS algorithm. Importantly, the number of these pheromone-update messages is proportional to the length of the RMST itself (where the pheromone is deposited), as there is no need to update on zero pheromone values elsewhere. In a sense, this message type implements bounded dissipation—and its relatively high communication overhead is the reason why full dissipation would be very costly in an artificial communication network (while ants use it for free in nature).

The distributed Prim's algorithm involves flood messages among the impact nodes in P, updates of the global hierarch, and most importantly, the updates of neighbours of each cell on the values of the multiple gradient fields stored in the cell. Again, this last message type is the main component of the communication cost. The aggregate cost of these gradient updates is so high because it is dependent on the total number of cells in the array. This analysis shows that the ACO-DRS algorithm will scale much better in terms of the communication cost, with the increase in the array's size, as the length of the RMST in any array is approximately an order of magnitude smaller than the number of cells in the array. In addition, the model involving a global-hierarch is inherently non-robust, and will be less robust as the size increases.

7. Conclusion and future directions

This paper presented a novel application of ACO and gradient-based algorithms to the problem of finding minimum spanning trees in a dynamic decentralised setting, exemplified by self-monitoring and self-repairing aerospace vehicles. The new localised algorithms described here are based on the ant colony metaphor and lead to emergence of reconfigurable impact networks. These networks represent a broader class and demonstrate the potential of self-organising multi-agent networks in detecting damaged sections of the modular multi-cellular vehicle skin. The investigation of dynamics of produced rectilinear minimum spanning trees complements the developed methods, and allows us to quantitatively evaluate the stability of the emergent patterns. In particular, a metric based on the variance in the size of the LCS captures connectivity in impact networks, pinpointing important phase transitions.

We have also evaluated the ACO-DRS algorithm against a variant from the class of distributed dynamic programming algorithms, using Prim's algorithm periodically invoked in a global hierarch. The comparative analysis highlighted a low communication overhead of the ACO-DRS algorithm, and a better quality of RMST approximations computed by the alternative distributed Prim's algorithm. The higher communication cost of the alternative variant may be acceptable if the quality of RMST-approximation is a serious issue, while robustness can be increased by replacing the global-hierarch with a network of local hierarchs. The role of local hierarchs can be taken by the impact nodes (the points of interest) themselves. We are currently developing such a hybrid algorithm without a global hierarch, where ant-like messages are used for discovering the impact nodes and a gradient-like field is used for minimising the edge costs. The communication cost of this new algorithm is expected to be lower than that of the distributed Prim's algorithm, although still higher than that of the ACO-DRS algorithm. In summary, there is a range of algorithmic options, and their choice depends on priorities of the task at hand. We would like to point out that in a distributed decentralised setting communication complexity is often more important than the computation complexity.

A parallel study [21,36,30,31] describes an analysis of spatiotemporal stability of impact boundaries self-organising around (sub-)critical impacts, and their use in multi-cellular shape replication. The results presented there identify phase transitions in terms of generic information-theoretic properties, such as the

Shannon entropy, and form a part in our overall methodology, contributing to global response engineering in multi-agent networks. A key step of this methodology is a clear identification of appropriate selection pressures (such as stability of emergent patterns, communication cost, etc.), the corresponding metrics, and well-performing ranges within the parameter space, employed by the localised algorithms. This opens a way towards *evolvable* algorithms for reconfigurable sensor and communication networks.

Acknowledgments

This work was carried under CSIRO contract to NASA. It is a pleasure to record our appreciation to Dr Ed Generazio (NASA Langley Research Center) for his encouragement and support. The authors are grateful to other members of the AAV project for many open and motivating discussions. Mark Foreman's summer vacation scholarship was provided by CSIRO Telecommunication and Industrial Physics, and we would like to thank Dr Geoff James in particular.

References

[1] D. Abbott, B. Doyle, J.B. Dunlop, A.J. Farmer, M. Hedley, J. Herrmann, G. James, M.E. Johnson, B. Joshi, G.T. Poulton, D.C. Price, M. Prokopenko, T. Reda, D.E. Rees, D.A. Scott, P. Valencia, D. Ward, J. Winter, Development and evaluation of sensor concepts for Ageless Aerospace Vehicles, Development of concepts for an intelligent sensing system, Technical Report NASA/CR-2002-211773, NASA Langley Research Center, Hampton, VA, 2002.

[2] D. Abbott, J. Ables, A. Batten, D.C. Carpenter, A.F. Collings, B. Doyle, J.B. Dunlop, G.C. Edwards, A.J. Farmer, B. Gaffney, M. Hedley, P. Isaacs, M.E. Johnson, B. Joshi, C.J. Lewis, G.T. Poulton, D.C. Price, M. Prokopenko, T. Reda, D.E. Rees, D.A. Scott, S. Seneviratne, P. Valencia, P. Wang, D.F. Whitnall, J. Winter, Development and evaluation of sensor concepts for Ageless Aerospace Vehicles, Report 3: Design of the Concept Demonstrator, Confidential Report No. TIPP 1628, CSIRO Telecommunications and Industrial Physics, Australia, January 2003.

[3] A. Batten, J.B. Dunlop, G. Edwards, A.J. Farmer, B. Gaffney, M. Hedley, N. Hoschke, P. Isaacs, M.E. Johnson, C.J. Lewis, A. Murdoch, G.T. Poulton, D.C. Price, M. Prokopenko, I. Sharp, D.A. Scott, P. Valencia, P. Wang, D. Whitnall, Development and evaluation of sensor concepts for Ageless Aerospace Vehicles, Report 5: Phase 2—Implementation of the Concept Demonstrator, Confidential Report No. TIPP 2056, CSIRO Telecommunications and Industrial Physics, Australia, April 2004.

[4] H. Bojinov, A. Casal, T. Hogg, Multiagent control of self-reconfigurable robots, Artif. Intell. 142 (2002) 99–120.

[5] E. Bonabeau, G. Theraulaz, J.-L. Deneubourg, S. Camazine, Self-organisation in social insects, Trends Ecol. Evol. 12 (5) (1997) 188–193.

[6] D. Braginsky, D. Estrin, Rumor routing algorithm for sensor networks, Proceedings of the First ACM International Workshop on Wireless Sensor Networks and Applications, Atlanta, GA, 2002, pp. 22–31.

[7] J. Buhl, J.L. Deneubourg, G. Theraulaz, Self-organized networks of galleries in the Ant Messor Sancta, in: M. Dorigo, G. Di Caro, M. Sampels (Eds.), Proceedings of the Third International Workshop on Ant Algorithms, ANTS 2002, Brussels, Belgium, 2002, pp. 163–175.

[8] A. Colorni, M. Dorigo, V. Maniezzo, Distributed optimization by ant colonies, in: F.J. Varela, P. Bourgine (Eds.), Toward a Practice of Autonomous Systems: Proceedings of the First European Conference on Artificial Life, Paris, December 1991, A Bradford Book, MIT Press, Cambridge, MA, 1992, pp. 134–142.

[9] O. Cordon, F. Herrera, T. Stützle, A review on the ant colony optimization metaheuristic: basis, models and new trends, Mathware Soft Comput 9 (2/3) (2002) 141–175.

[10] A. Das, R.J. Marks II, M.A. El-Sharkawi, P. Arabshahi, A. Gray, The minimum power broadcast problem in wireless networks: an ant colony system approach, Proceedings of IEEE CAS Workshop on Wireless Communications and Networking, Pasadena, CA, 2002.

[11] J.L. Deneubourg, S. Goss, Collective patterns and decision making, Ethol. Ecol. Evol. 1 (1989) 295–311.

[12] G. Di Caro, M. Dorigo, AntNet: distributed stigmergetic control for communications networks, J. Artif. Intell. Res. 9 (1998) 317–365.

[13] M. Dorigo, V. Maniezzo, A. Colorni, The Ant System: an autocatalytic optimising process, Technical Report No. 91-016 (Revised), Politecnico di Milano, Milan, Italy, 1991.

[14] M. Dorigo, V. Maniezzo, A. Colorni, The Ant System: optimization by a colony of cooperating agents, IEEE Trans. Syst. Man Cybern. B 26 (1) (1996) 1–13.

[15] M. Dorigo, L.M. Gambardella, Ant colonies for the traveling salesman problem, BioSystems (1997).

[16] M. Dorigo, L.M. Gambardella, Ant colony system: a cooperative learning approach to the traveling salesman problem, IEEE Trans. Evol. Comput. 1 (1) (1997) 53–66.

[17] M. Dorigo, G. Di Caro, Ant Colony Optimization: a new metaheuristic, in: Proceedings of the 1999 Congress on Evolutionary Computation, Washington, DC, IEEE Press, Piscataway, NJ, 1999, pp. 1470–1477.

[18] P. Erdös, A. Rényi, On the strength of connectedness of random graphs, Acta Math. Sci. Hungary 12 (1961) 261–267.

[19] D. Estrin, R. Govindan, J. Heidemann, S. Kumar, Next century challenges: scalable coordination in sensor networks, in: H. Kodesh, V. Bahl, T. Imielinski, M. Steenstrup (Eds.), Proceedings of the Fifth Annual ACM/IEEE International Conference on Mobile Computing and Networks, Seattle, ACM Press, New York, NY, 1999, pp. 263–270.

[20] D. Eppstein, Spanning trees and spanners, Technical Report 96-16, ICS, UCI, 1996; also in: J.-R. Sack, J. Urrutia (Eds.), Handbook of Computational Geometry, Elsevier, 1999, pp. 425–461.
[21] M. Foreman, M. Prokopenko, P. Wang, Phase transitions in self-organising sensor networks, in: W. Banzhaf, T. Christaller, P. Dittrch, J.T. Kim, J. Ziegler (Eds.), Advances in Artificial Life—Proceedings of the Seventh European Conference on Artificial Life (ECAL), Dortmund, Springer-Verlag, Heidelberg, Germany, 2003, pp. 781–791.
[22] D. Green, Emergent behavior in biological systems, in: D.G. Green, T.J. Bossomaier (Eds.), Complex Systems—From Biology to Computation, IOS Press, Amsterdam, The Netherlands, 1994, pp. 24–35.
[23] A.B. Kahng, G. Robins, On Optimal Interconnections for VLSI, Kluwer Academic Publishers, Boston, MA, 1995.
[24] A.S. Klyubin, D. Polani, C.L. Nehaniv, Organization of the information flow in the perception-action loop of evolved agents, Proceedings of the 2004 NASA/DoD Conference on Evolvable Hardware, IEEE Computer Society, 2004.
[25] C. Langton, S. Forest (Eds.), Computation at the Edge of Chaos: Phase Transitions and Emergent Computation, MIT Press, Cambridge, MA, 1991, pp. 12–37.
[26] H. Lovatt, G.T. Poulton, D.C. Price, M. Prokopenko, P. Valencia, P. Wang, Self-organising impact boundaries in ageless aerospace vehicles, in: J.S. Rosenschein, T. Sandholm, M. Wooldridge, M. Yokoo (Eds.), Proceedings of the Second International Joint Conference on Autonomous Agents and Multi-Agent Systems, Melbourne, ACM Press, New York, NY, 2003, pp. 249–256.
[27] R. Nagpal, Programmable self-assembly using biologically-inspired multiagent control, Proceedings of the First International Joint Conference on Autonomous Agents and Multi-Agent Systems, Bologna, Italy, 2002, pp. 418–425.
[28] K.C. Prevas, C. Ünsal, M.Ö. Efe, P.K. Khosla, A hierarchical motion planning strategy for a uniform self-reconfigurable modular robotic system, Proceedings of IEEE International Conference on Robotics and Automation, 2002, pp. 787–792.
[29] R.C. Prim, Shortest connection networks and some generalizations, Bell Syst. Techn. J. 36 (1957) 1389–1401.
[30] M. Prokopenko, P. Wang, On self-referential shape replication in robust aerospace vehicles, Proceedings of the Ninth International Conference on the Simulation and Synthesis of Living Systems (ALIFE9), Boston, USA, 2004.
[31] M. Prokopenko, P. Wang, D.C. Price, P. Valencia, M. Foreman, A.J. Farmer, Self-organising hierarchies in sensor and communication networks, Artif. Life 11 (4) (2005) 1–20.
[32] M. Prokopenko, P. Wang, D. Price, Complexity metrics for self-monitoring impact sensing networks, in: Proceedings of 2005 NASA/DoD Conference on Evolvable Hardware (EH-05), Washington, DC, USA, 2005, pp. 239–246.
[33] R. Schoonderwoerd, O.E. Holland, J.L. Brutten, L.J.M. Rothkrantz, Ant-based load balancing in telecommunication networks, Adapt. Behav. 5 (2) (1997) 169–207.
[34] R.V. Solé, S. Valverde, Information theory of complex networks: on evolution and architectural constraints, in: E. Ben-Naim, H. Frauenfelder, Z. Toroczkai (Eds.), Complex Networks, Lecture Notes in Physics, vol. 650, Springer, 2004.
[35] P. Wang, P. Valencia, M. Prokopenko, D.C. Price, G.T. Poulton, Selfreconfigurable sensor networks in ageless aerospace vehicles, in: U. Nunes, A.T. de Almeida, A.K. Bejczy, K. Kosuge, J.A.T. Machado (Eds.), IEEE Proceedings of the 11th International Conference on Advanced Robotics (ICAR), Coimbra, Universidade de Coimbra 2003, Coimbra, Portugal, 2004, pp. 1098–1103.
[36] P. Wang, P. Valencia, M. Prokopenko, Evolvable recovery membranes in self-monitoring aerospace vehicles, Proceedings of the Eighth International Conference on Simulation of Adaptive Behaviour (SAB-2004), Los Angeles, USA, 2004, pp. 509–518.
[37] R. Wehner, M.V. Srinivasan, Searching behaviour of desert ants, genus *Cataglyphis* (Formicidae, Hymenoptera), J. Comput. Physiol. 142 (1981) 315–338.
[38] J. Wessnitzer, A. Adamatzky, C. Melhuish, Towards self-organising structure formations: a decentralized approach, in: J. Kelemen, P. Sosík (Eds.), Proceedings of the Sixth European Conference on Advances in Artificial Life, Prague, Czech Republic, 2001, pp. 573–581.
[39] S. Wolfram, Universality and complexity in cellular automata, Physica D 10 (1984) 1–35.
[40] A. Wuensche, Classifying cellular automata automatically: finding gliders, filtering, and relating space-time patterns, attractor basins, and the *Z* parameter, Complexity 4 (3) (1999) 47–66.
[41] Y.-F. Wu, P. Widmayer, M.D.F. Schlag, C.K. Wong, Rectilinear shortest paths and minimum spanning trees in the presence of rectilinear obstacles, IEEE Trans. Comput. C-36 (1987) 321–331.
[42] F. Ye, A. Chen, S. Lu, L. Zhang, A scalable solution to minimum cost forwarding in large scale sensor networks, Proceedings of the 10th International Conference on Computer Communications and Networks, Dallas, Texas, 2001, pp. 304–309.

Mikhail Prokopenko is a Senior Research Scientist at the CSIRO ICT Centre, Autonomous Systems Laboratory. He holds a Ph.D. (Computer Science, Macquarie University, Australia), an M.A. (Economics, University of Missouri-Columbia, USA), and an M.S. (Applied Mathematics, Azerbaijan Oil and Chemistry Institute, USSR). His long-term research goal is design and control of complex self-organising multi-agent systems, and an effective integration of real-time perception, reasoning and action in autonomous agents. His work has resulted in over 50 publications and patents. In June 2002, Mikhail Prokopenko received the JSAI (Japanese Society for Artificial Intelligence) award for scientific contribution to the RoboCup Simulation League. Currently, he is coordinating the Multi-Agent Algorithms team within the CSIRO-NASA research project on Ageless Aerospace Vehicles.

Peter Wang graduated with a B.Sc. in Computer Science and Information Systems from Macquarie University (Australia) in 2000. He is working in the CSIRO ICT Centre, Autonomous Systems Laboratory, in the research and development area of autonomous agents and multi-agent systems. He has designed and developed multi-agent systems in a number of application areas, including RoboCup Simulation and reconfigurable sensor and communication networks (CSIRO-NASA project on Ageless Aerospace Vehicles). In June 2002, Peter Wang received the JSAI (Japanese Society for Artificial Intelligence) award for scientific contribution to the RoboCup Simulation League—for the work on entropy of joint beliefs as a measure of multi-agent coordination potential.

Mark Foreman has completed a Bachelor of Engineering with first class honours in Computer Systems Engineering, and a Bachelor of Science in Mathematics and Computer Science, both at the University of Adelaide. He undertook a vacation scholarship placement with the CSIRO over summer 2002/2003, contributing to research and development of quantitative metrics for phase transitions in multi-agent networks. During 2004, he commenced a Ph.D. in syntactic analysis with the CSIRO and the University of Adelaide.

Philip Valencia received his B.Eng. (Electronics) and BIT from Queensland University of Technology in 2000. Joining the CSIRO in 2001, Philip has gained extensive experience in complex systems, image processing and active vision, machine learning and artificial intelligence. His recent work has taken him to focus on robotics in heterogeneous sensor networks including actuation and mobile robots in self-managing ad hoc sensor networks and autonomous robotic repair of a multi-cellular ageless aerospace vehicle surface. He is presently facilitating CSIRO-wide interactions on Engineering, Design and Control of Complex Systems.

Don Price received B.Sc.(Hons) and Ph.D. degrees in physics from Monash University, Australia, in 1966 and 1970, respectively. Since joining the CSIRO in 1982, he has been involved in the development of ultrasonic transducers, ultrasonic imaging, studies of the physics of linear and non-linear elastic wave propagation in inhomogeneous materials (with particular emphasis in recent years on propagation in planar layered structures), and the development of ultrasonic measurement techniques based on these studies. He is currently the Chief Scientist for CSIRO Industrial Physics, and is the Project Leader of the Self-organising Sentient Structures project, a collaborative investigation, supported by CSIRO and the NASA Langley Research Centre, aimed at integrated smart sensing and communication systems for structural health management.

Geoff Poulton is currently a Senior Principal Research Scientist within the CSIRO ICT Centre in Sydney. He has had a long research career within Universities and CSIRO with his main areas of expertise being originally in electromagnetic theory and antennas, and latterly in intelligent systems, complex systems and biometrics. His most recent research interest has been in the area of design and control complex systems, a difficult task since such systems exist on the "edge of chaos". He has produced over 130 publications and patents, has been awarded two CSIRO Medals and has a long-standing interest in cross-disciplinary research. Dr. Poulton is also a Director of the Biometrics Institute.