Heterogeneous Multiagent Architecture for Dynamic Triage of Victims in Emergency Scenarios

Estanislao Mercadal, Sergi Robles, Ramon Martí, Cormac J. Sreenan and Joan Borrell

Abstract This paper introduces a double multiagent architecture allowing the triage of victims in emergency scenarios and the automatic update of their medical condition. Gathering updated information about the medical condition of victims is critical for designing an optimal evacuation strategy that minimizes the number of casualties in the aftermath of an emergency. The proposed scheme, currently under development, combines Wireless Sensor Networks, an Electronic Triage Tag and a double multiagent system (Agilla-JADE) to achieve a low cost, no infrastructure-based, efficient system.

1 Introduction

The first moments of a mass casualty incident aftermath are decisive to save the highest number of lives. Only a reduced number of resources are often available at this point, making it essential to coordinate efforts and evacuate and give urgent treatment to the most seriously injured yet curable victims. Thus, the field work of trained personnel (doctors, nurses, paramedical) triaging victims accordingly to their medical status is of paramount importance. The triage information is later used to decide who are the victims to be moved first. For many years, this has been achieved by using a cardboard triage tag (like the one depicted in Fig. 1) which includes an easily identifiable color code indicating the severity of the injuries, besides other basic medical information. Triage personnel follows a standard expedited triage method like MTS [10] or START [17].

Estanislao Mercadal, Sergi Robles, Ramon Martí, and Joan Borrell
Department of Information and Communications Engineering, Universitat Autònoma de Barcelona, 08193 Cerdanyola del Vallès, Spain, e-mail: {emercadal, srobles, rmarti, jborrell}@deic.uab.cat
Cormac J. Sreenan
Departament of Computer Science, University College Cork, Cork, Ireland, e-mail: cjs@cs.ucc.ie

It is obvious that these scenarios can take a clear advantage of Information and Communication Technologies to improve efficiency, although some strong limitations have to be taken into account. Emergency personnel, for instance, have to act quickly and will refuse to interact with complex systems or fill in forms *in situ*. Communications, on the other hand, should not rely on any local infrastructure. Existing infrastructure (wired or not) can be damaged or out of order, and setting up a new one can be very expensive in terms of money and/or time.

Mobile Agent based Electronic Triage Tag (MAETT) [11] is an interesting agent-based application showing how mobile agents [3] can significantly help in this type of scenarios while keeping the budget low. A touch screen handheld device, equipped with a GPS receiver, wireless network interface and a JADE [1] mobile agent platform, is the key element of the system. When a victim is evaluated in the triage stage, a mobile agent is created holding the START color code, GPS position and other information. The traditional cardboard triage tag is also placed on the victim. This agent will make its way to the emergency coordination center, leaping forward from device to device. It may be stored in a handheld device for a while, being physically carried by a member of the emergency personnel, waiting until a candidate device is at reach. The network created in this fashion goes beyond *ad hoc* or MANET possibilities, for no concomitant communication links are required from the source device to the destination. The routing protocol uses the estimated time to return to the coordination center of the handheld device bearer.

MAETT works well, and makes information about victims available to the coordination center at a low cost. This information is later used to plan the evacuation of the victims. Unfortunately, changes in victims' medical conditions, which have a great impact on the subsequent rescue planning, are never conveyed to the coordination center. The solution to this is not trivial; the new information is not triggered by the field personnel but by the victims. The chances for personnel of finding a victim are much higher than the other way around.

Our final goal is to devise a new dynamic electronic triage tag system where a variation in the medical situation of victims can be easily communicated to the coordination center. This dynamism will be achieved by placing a wireless node equipped with medical sensors in every victim, in addition to the triage tag in the MAETT scenario. These nodes run an Agilla [5] mobile agent platform and can communicate with each other if they are near by forming a Wireless Sensor Network (WSN) [4]. The nodes monitor the victim's vital signs, and communicate any significant change to the emergency personnel nearby, should they are in range. The nodes in the WSN are very limited in resources, and cannot be compared to the devices carried by the emergency personnel. Thus, communication strategies, carried data and code differ completely from the network created in MAETT, and an interface is required to connect both.

To improve the probability of sending information from a node in the WSN to a handheld device, an efficient and adaptable mechanism based on mobile Agilla agents and genetic algorithms is currently under development, to explore all the nodes of the WSN and to fuse all the information about changes on victims' status. As a result, any node in the WSN will be able to communicate these changes as a whole to any JADE platform (triage or rescue member) that moves into the wireless coverage area of any node in the WSN. JADE mobile agents will act as data mules [16] for the independent WSNs created in the emergency area, and will carry to the coordination center the initial classification of the victims and any significant subsequent change in their medical status.

2 Background

2.1 MAETT

MAETT (Mobile Agent Electronic Triage Tag) [11] is a system providing early resource allocation during emergencies when no network infrastructure is available.

The foundation of the system is mobile agent technology [3], which allows information to be directly transported from terminal to neighbor terminal regardless of the status of the rest of the network at that particular time. Handheld devices run an execution environment for agents, the platform, where mobile agents can be created, executed and forwarded to other terminals. Are the agents themselves who decide the route to follow depending upon the available information on the neighbors.

The main actors of the system are the victims, the triage personnel, and the rescue teams (see scenario in Fig. 1). Victims are supposed to be scattered over an arbitrarily large area of emergency. The triage personnel scour all this area looking for victims and triage them according to standard methods. The result of this triage is written in a physical tag and placed visibly on the victim. Finally, the rescue teams collect all the victims, prioritizing depending on triage results. The Emergency Coordination Center (ECC) coordinates all actions. Triage personnel, rescue teams and the ECC have wireless devices with a JADE mobile agent platform and a GPS receiver.

Triage personnel leave the ECC, and have an estimation on when they will get back (TTR). When a victim is found, they use the standard START method [17] and place a cardboard triage tag (see Fig. 1) on the neck of the victim with their evaluation written on it. The tag has an integrated RFID. At the same time, an agent is created containing the information in the tag, plus the GPS position of the victim and the RFId of the tag. All this information will be used later in the ECC to optimize the route of the rescue teams. This agent is transmitted to neighbor devices only if the bearer has a lesser TTR. This is to make sure that moving the information is never going to make it arrive later. Consequently, all handheld devices carried by triage personnel are used to create agents with information about found victims, and also to forward agents corresponding to other victims. When agents arrive to the ECC, the ECC send the rescue teams with a detailed schedule of the route based on the GPS position of victims as well as their medical condition.

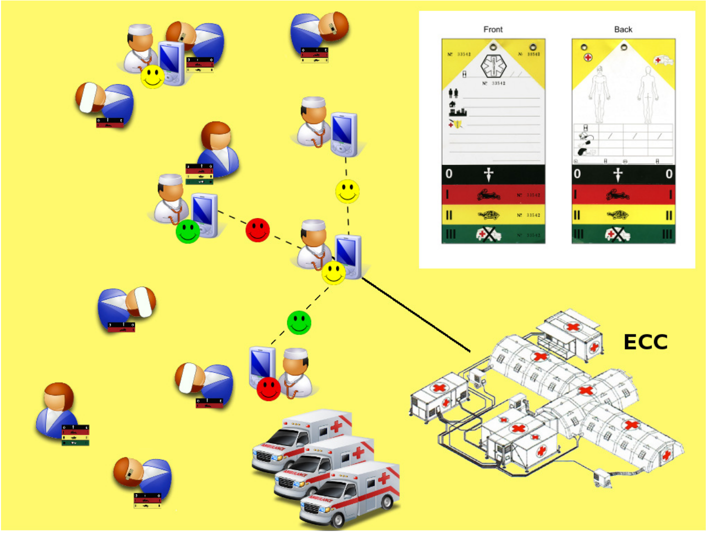


Fig. 1 MAETT triaging scenario showing also the classical cardboard triage tags

2.2 Wireless Sensor Networks and Agilla mobile agent middleware

A Wireless Sensor Network (WSN) [4] is a specific type of ad hoc networks, built using wireless radio communication. It consists of sensor nodes collecting particular measures, i.e. sensor readings, and processing elements, which collect these measures for further processing. Usually sensor networks are strongly resource-restricted in terms of communication, processing and storage capabilities, and in terms of available energy. For the most part, all sensor nodes deliver their data to a base station or *sink* of data. This *sink* can be part of the network, or be external, accessed trough a gateway to other networks.

Application examples of WSNs [4, 5] are increasing, and include, among others, emergency operations, habitat monitoring, precision agriculture, home automation and health care or logistics. Initial applications within WSNs were static, i.e. all the nodes run statically installed software loaded prior to their deployment. Nowadays, diferent systems have been developed to allow more adaptable WSN applications (see [5] for an excellent survey of such systems), from mechanisms that allow a total or partial reprogrammation of the nodes, to middlewares that allow the execution of several mobile agents inside each node, allowing application self-adaptability. Among the different mobile agent middlewares proposed for WSNs, Agilla [5] is the first one deployed inside real WSNs.

Agilla provides a programming model in which applications consist of evolving communities of agents that share a WSN. Agents can dynamically enter and exit the WSN, can autonomously clone and migrate themselves in response to environmental changes, and can mantain a global coordination. Agilla was implemented on top of TinyOS WSN operating system [6], and experimentally evaluated on several real WSNs, for instance those consisting of TelosB [14] nodes.

3 Heterogeneous multiagent architecture to provide dynamism to MAETT

Despite being two different agent technologies, JADE and Agilla can coexist and share information to build a more complex agent system. Albeit agents themselves can not migrate transparently between the two platforms, little changes in Agilla's code make this cooperation possible.

In our solution, we use Agilla to continuously monitor Mass Casualty Incident (MCI) victims inside WSNs and JADE to carry the monitored data to the Emergency Coordination Center (ECC), introducing dynamism to MAETT. We take advantage of the communication between both technologies to share victim information and route details, thus improving the efficiency of the triaging system.

We extend MAETT scheme (Fig. 1) by adding to each victim a wireless human body monitoring device, for example that of Fig. 2(a), a TelosB compatible node manufactured by Maxfor (http://maxfor.co.kr), thus creating a WSN among neighboring victims, which we can use to dynamically update the medical status of every victim.

To communicate the WSN and the handheld devices of triage or rescue members we also need these members carry a WSN node (Fig. 2(b)) attached to their handheld device, also a TelosB compatible node from Maxfor.

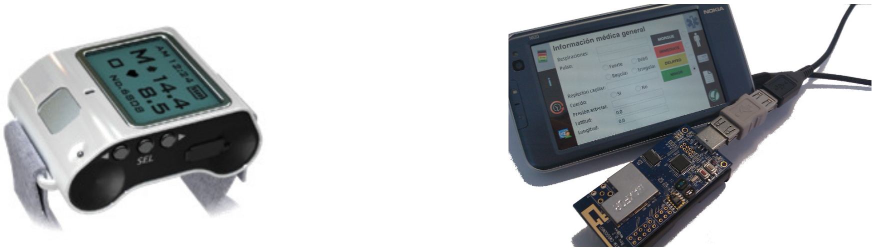
3.1 WSN set up and operation

In the first run of the triage personnel, every victim receives the classical triage tag, and it is also equipped with a wireless body monitoring sensor node. Inside this node there is an Agilla monitor agent with the specific data of the patient (Victim ID and Medical Condition). Each of this monitor agents continuously supervise the associated victim health constants mantaining an up-to-date log with his/her medical condition. In turn, the wireless body monitoring sensor node is identified by a two dimensional coordinate (WId, AId), the first one (WId) indicates its WSN and the second one (AId) identifies the node into the network.

As every nearby victim is both paper tagged and electronically tagged, neighboring wireless nodes belonging to the same triage member wirelessly connect creating a growing WSN of victims.

When every victim in the vicinity is tagged and every body sensor node monitors its own victim, the triage team member ends the creation of his WSN. First of all, the handheld device starts the calculation of a genetic algorithm following the appproach of [12, 18] (see Section 3.2) to compute a good enough itinerary to roam the newly created WSN to fetch the updates of every victim's medical condition. A roaming agent with this computed itinerary is then injected inside the last node of the WSN. This roaming agent gets and sets the differences in the state on every node.

When a new rescue or triage team member approaches the WSN, and their handheld device contacts a node in the WSN, the sensor node attached to the handheld device automatically identifies itself as the *sink* node. Then a mobile agent containing the information of every victim is sent to the node on the handheld device to flush all the gathered data. As every node of the WSN has the most up-to-date state as possible, any member of the WSN in direct contact with the *sink* node can indistinctly route the information. The handheld device then creates a JADE mobile agent containing the new status of the victims, and routes it to the ECC, as was done in MAETT.



(a) Watch type body monitoring device

(b) A sensor node connected to a handheld

Fig. 2 The different devices of the scenario.

To be able to use any node of the WSN to flush the status of every victim to the handheld device, every node must mantain a record of every other node status. This status is identified for each victim by her concrete medical condition, the ID of the victim, and a modification flag. The roaming mobile agent continuously updates this information mantaining an up-to-date record of every victim.

3.2 *Roaming the whole WSN*

The problem of visiting every node avoiding repetition and optimizing distance is deeply studied in graph theory, also known as the Travelling Salesman Problem, and is known to be NP-Complete [13] even in the Euclidean plane.

The case of covering all the nodes of a WSN can be seen as a particular case of the Travelling Salesman Problem, with the particularity of pretending to also optimize the energy consumption, and is also proven to be NP-Complete [18] [12].

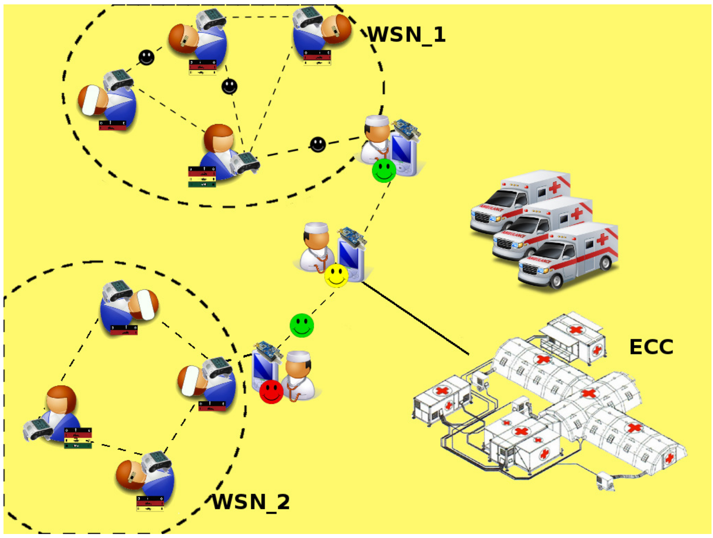


Fig. 3 Dynamic MAETT triaging scenario

Genetic algorithms are being used to solve NP-Complete problems since the early 90's [7] and have been proved useful to efficiently solve the concrete particular of the Travelling Salesman Problem [2] [15] [8].

In our handheld device, with its restrictions, both of time and computation power, is not feasible to calculate the optimal route for a problem of this kind. Thus, we use a highly studied genetic algorithm approximation [12], and which is proved to offer good enough solutions. Albeit not being the optimal solution, it returns a route which is satisfactory both in computation time and roaming time.

The chosen genetic algorithm starts creating an initial population of random paths that cover the network. Then, from that set of paths only those whose number of hops to cover the network is smaller or equal to the time remaining to meet the time deadline (maximum number of nodes per path) are selected. After that, a new generation is generated by selecting x individuals to be crossed. A subset of the paths of two individuals are crossed to create a new individual. Finally, all solutions are evaluated and sorted from the minimum to the maximum. We are currently adapting this genetic algorithm not to calculate the minimum route to a node, but to determine an efficient way to cover the whole WSN.

Additionally, if a problem in the communication between WSN nodes occur, e.g. due to a malfunctioning or broken node, the roaming agent has the ability to reroute itself and find an alternative path to cover the whole WSN. In our case the mobile agent tries to move to the next available node in the WSN. If it is not reachable tries to route the network backwards.

Note that the computation of the genetic algorithm in the handheld device is possible due to the following facts:

1. The triage personnel saves the topology of the WSN while tagging victims. When finished the device calculates a genetic algorithm adapted to that particular topology, and injects an agent with the final algorithm into the WSN. The Agilla agent then covers the whole WSN collecting and informing changes in victim's state, thus writing the information of every victim in every node. The agent keeps doing the same route keeping the victim's medical condition up-to-date.
2. The number of nodes is limited. Both to speed up the route calculation process and to lighten the medical personnel bags. Notice that every health monitoring sensor weights about 70g including the required two 1.5V batteries. Assuming that the triaging personnel carries the handheld device and the paper tags, they may agree to carry the extra weight of 20 to 25 sensor nodes, that is, the weight of a small laptop (i.e. 1.5Kg).

3.3 WSN maintenance

Another addition that boosts the dynamism of our triaging method is the possibility to modify an already deployed WSN, such that a victim can be added or removed, while being added to or removed from the itinerary of the roaming agent. Using JADE agents and their routes to the ECC, we have an excellent infrastructure to attain this dynamism.

In every triage, the handheld device of the triage personnel is also loaded with the position of every node of the WSN, thus ending up with a full view of the topology of the network. After the computation of the genetic algorithm this topology is also loaded into a JADE agent and routed to the ECC.

On arrival to the ECC, the JADE agent flushes the WSN topology, which is shared to every other handheld device deployed in the emergency. Hence if some other triaging member has to tag or remove a victim in another's WSN, he just removes or adds this new victim to the corresponding topology in the handheld device, and calculates a new itinerary (using our genetic algorithm) with the modification.

Of course, just as in the first calculation, the new topology is routed, along with the medical data of the victims in the WSN, with a JADE agent to the ECC.

3.4 The WSN - Handheld device interface

To properly set the interface we had to modify Agilla to run side-by-side with JADE. Doing so Agilla can obtain the instance of the running platform and generate an agent there. Then, we modified Agilla's AgentInjector to switch on JADE when invoked, this way we are sure that from that moment on, every recieved agent will be able to generate a JADE agent. The agent which migrates or sends data to the handheld device, the one to be encapsulated into a JADE agent, needs a special portion of code permitting the communication with the Injector at the other side of the USB interface. This code moves or sends the data to a special address pertaining to the Injector in the handheld device.

Our test platform (Nokia N810 with the Maemo OS (Diablo 5.2008.43.7)) did not come with the needed drivers for the mote to work. Thus, we had to recompile the stock kernel adding the appropiate drivers, and load them into the device. Moreover, albeit the device supports *host mode usb*, not every usb cable is prepared to notify the host device of this requirement, thus, we had to explicitly tell our test platform to switch itself to *host mode*. Once our test platform was plugged and the drivers loaded, the system recognized a new usb device, but complained about not being able to access it. The problem was related to the power the device was recieving not being sufficent to mantain it operating. We had to add a rule to the *udev* daemon of the handheld device indicating that this particular device needs more power than the allocated by default.

4 Conclusion

The triage stage during the aftermath of an emergency is critical to minimize the number of casualties. There are many systems in the bibliography introducing Information and Communication Technologies to achieve this, most times at a high cost or requiring a non-practical interaction with the system that field personnel refuse to perform. This paper presents a multiagent architecture allowing the triage of victims in emergency scenarios and the automatic update of their medical condition at a low cost.

In our approach, neighbor victims are automatically grouped together by placing a wireless sensor node in each of them monitoring their status. This groups form a Wireless Sensor Network. Changes are shared within the WSN using Agilla mobile agents and a gathering algorithm designed using genetic algorithms. Now, changes can be communicated to any member of the emergency personnel by any node of the group, increasing in this way the probabilities of sending this information. From there, a JADE mobile agent will carry the changes to the coordination center where they will update old information.

Currently we are working on the implementation, and simulating different routing strategies using *TOSSIM* and *TinyViz* [9] to determine the number of agents needed to cover the WSN as fast as possible and therefore propagating the changes in victim's state rapidly.

Acknowledgements

This work has been funded by the Spanish Ministry of Science and Innovation through the project TIN2010-15764.

References

1. F.L. Bellifemine, G. Caire, and D. Greenwood. *Developing multi-agent systems with JADE*. Wiley Inc., 2007.
2. H. Braun. On solving travelling salesman problems by genetic algorithms. In Hans Paul Schwefel and Reinhart Männer, editors, *Parallel Problem Solving from Nature*, volume 496 of *Lecture Notes in Computer Science*, pages 129–133. Springer Berlin / Heidelberg, 1991. 10.1007/BFb0029743.
3. J. Cucurull, J. Ametller, and R. Martí. Agent mobility. In *Developing multi-agent systems with JADE*, pages 115–130. Wiley Inc., 2007.
4. F. Dressler. *Self-Organization in Sensor and Actor Networks*. Wiley Inc., 2007.
5. Ch-L. Fok, G-C. Roman, and Ch. Lu. Agilla: A mobile agent middleware for self-adaptive wireless sensor networks. *ACM Trans. Auton. Adapt. Syst.*, 4(3):1–26, 2009.
6. J. Hill, R. Szewczyk, A. Woo, S. Hollar, D. Culler, and K. Pister. System architecture directions for networked sensors. In *Proceedings of the ninth international conference on Architectural support for programming languages and operating systems*, pages 93–104. ACM, 2000.
7. K.A. De Jong and W.M. Spears. Using genetic algorithms to solve NP-complete problems, 1989.
8. P. Larrañaga, C.M.H. Kuijpers, R.H. Murga, I. Inza, and S. Dizdarevic. Genetic algorithms for the travelling salesman problem: A review of representations and operators. *Artificial Intelligence Review*, 13:129–170, 1999. 10.1023/A:1006529012972.
9. P. Levis, N. Lee, M. Welsh, and D. Culler. TOSSIM: accurate and scalable simulation of entire tinyos applications. In *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 126–137, New York, NY, USA, 2003. ACM.
10. K. Mackway-Jones, editor. *Emergency triage, 2nd ed.* Wiley Inc., 2006.
11. R. Martí, S. Robles, A. Martín-Campillo, and J. Cucurull. Providing early resource allocation during emergencies: The mobile triage tag. *Journal of Network and Computer Applications*, 32:1167–1182, 2009.
12. D. Massaguer. Multi mobile agent deployment in wireless sensor networks. Master's thesis, University of California, Irvine, 2005.
13. C.H. Papadimitriou. The Euclidean travelling salesman problem is NP-complete. *Theoretical Computer Science*, 4(3):237 – 244, 1977.
14. J. Polastre, R. Szewczyk, and D. Culler. Telos: Enabling ultra-low power wireless research. In *IPSN '05 Proceedings of the 4th international symposium on Information processing in sensor networks*, pages 364–369. ACM and IEEE, 2005.
15. J-Y. Potvin. Genetic algorithms for the traveling salesman problem. *Annals of Operations Research*, 63:337–370, 1996. 10.1007/BF02125403.
16. R.C. Shah, S. Roy, S. Jain, and W. Brunette. Data mules: Modeling a three-tier architecture for sparse sensor networks. In *Proc. Sensor Network Protocols and Applications (SNPA)*, pages 30 – 41. IEEE, May 2003.
17. G. Super. START: a triage training module. Newport Beach, California: Hoag Memorial Hospital Presbyterian, 1984.
18. Q. Wu, N.S.V. Rao, J. Barhen, S.S. Iyengar, V.K. Vaishnavi, H. Qi, and K. Chakrabarty. On computing mobile agent routes for data fusion in distributed sensor networks. *IEEE Trans. on Knowl. and Data Eng.*, 16(6):740–753, 2004.