Multi-Agent System for Security Control on Industrial Environments

Dante I. Tapia, Juan F. De Paz, Sara Rodríguez, Javier Bajo and Juan M. Corchado

Departamento de Informática y Automática, Universidad de Salamanca
Plaza de la Merced s/n, 37008, Salamanca, España
Email: {dantetapia, fcofds, srg, jbajope, corchado}@usal.es
http://bisite.usal.es/

Abstract: This paper presents a multi-agent system for managing and monitoring surveillance routes for security guards on industrial environments. The system obtains automatic and real-time information about the context to schedule and monitor the security guards activities. Users interact with the system in a simple, natural and intuitive way, using a set of wireless technologies and software agents with reasoning and planning mechanisms.

Keywords: Industrial Security, Agents, Surveillance Routes Calculation, Monitoring, Radio-Frequency Identification.

1. Introduction

There are plenty agents-based architectures, evolved in part because of the advances on intelligent environments and computational networks [12]. This has encouraged the development of ubiquitous computing [8], which constitutes the most optimistic approach to solve the challenge to create strategies that allow the anticipation and prevention of problems on automated environments [1].

Agents and multi-agent systems are progressively relevant in the development of dynamic and distributed systems, successfully implemented in areas such as e-commerce, medicine, oceanography, robotics, etc. [3]. The use of wireless technologies, such as GPRS (General Packet Radio Service), UMTS (Universal Mobile Telecommunications System), Bluetooth, etc., make possible to find better ways to provide mobile services and also give the agents the ability to communicate using portable devices (e.g. PDA's and cellular phones) [12].

This paper presents a multi-agent system aimed to automatically manage and monitor surveillance routes for security guards on industrial environments. The system uses a set of wireless technologies: Wi-Fi, GPRS (General Packet Radio Service) and Radiofrequency Identification (RFID). These technologies provide the structure to support the agents distributed communication needs. Moreover, they increase the mobility, flexibility and efficiency of users, allowing them to access resources (programs, equipment, services, etc.) remotely, no matter their physical location.

Next, the problem description related to monitoring workers activities on industrial environments is presented. Then, a case study is defined, describing the development of a multi-agent system designed to solve some of the problems that affect the industrial sector. Finally, the main characteristics of the system are explained, and the results and conclusions are presented.

2. Problem Description

The industrial sector has suffered an important evolution, from individual processes to management of virtual enterprises. But there are some aspects that still need some development, especially in techniques and technology to monitoring the workers activities in more efficient ways. In such an important and growing sector, it is necessary to establish security policies to manage risks and control hazardous events, providing better working conditions and an increase in productivity.

Recent studies [5] have revealed that at least 3% of the working shifts time is spent because of the lack of time control systems that supervise the real working time. Implementation of time control systems have a good influence in productivity, since the workers optimize their potential and enhance the process where they collaborate.

Multi-agent systems and intelligent mobile devices architectures have been recently explored as supervision systems, with the flexibility to be implemented in a wide diversity of scenarios, including industrial environments. These systems are suitable to handle complex and highly dynamic problems in execution time. They also offer robustness, scalability, high tolerance to failures, and coordination among their components [6], important features to be considered in the development of industrial systems [4].

Next, a case study is presented, describing the main technologies used to schedule and monitor security guards surveillance routes on industrial environments.

3. Case Study

A multi-agent system has been developed to improve some aspect related to security on industrial environments. This system allows scheduling and distribution of surveillance routes among available security guards. The routes assigned are automatically and real-time monitored to ensure the accomplishment of the security guards working shifts.

The system interacts with users through a set of mobile devices (PDA's) and wireless communication technologies (Wi-Fi, GPRS and RFID). These technologies and devices work in a distributed way, providing the users a flexible and easy access to the system services.

The agents in the system calculate the surveillance routes depending on the security guards available, the working shifts and the distance to be covered in the facilities. A supervisor (person) can set the possible routes, defining the areas that must be supervised, which can be modified according the scenario or changes in the environment. The system has the ability to re-plan the routes automatically considering the security guards available. It is also possible to track the workers activities (routes completion) over the Internet.

Radiofrequency Identification (RFID) is a key technology in this development. RFID is an automated data-capture technology that can be used to electronically identify, track, and store information about products, items, components or people. It is most frequently used in industrial/manufacturing, transportation, distribution, and warehousing industries, but there are other growth sectors including health care [13]. An RFID system contains basically four components: tags, readers, antennas and software. Tags with no power system integrated (e.g. batteries) are called passive tags or "transponders", these are much smaller and cheaper than active tags (power system included), but have shorter read range. The transponder is placed on the object itself (e.g. bracelet). As this object moves into the reader's capture area, the reader is activated and begins signalling via electromagnetic waves (radio frequency). The transponder subsequently transmits its unique ID information number to the reader, which transmit it to an end device or central computer where information is processed and delivered [13].

As can be seen on Fig. 1, the RFID configuration for the system presented in this paper consists of a mesh of tags distributed all over the building. Each tag, named "control point" is related to an area which must be covered by the security guards. Each security guard carries on a PDA with a RFID reader to register the completion of each control point. The information is sent via wireless to a central computer to be processed.

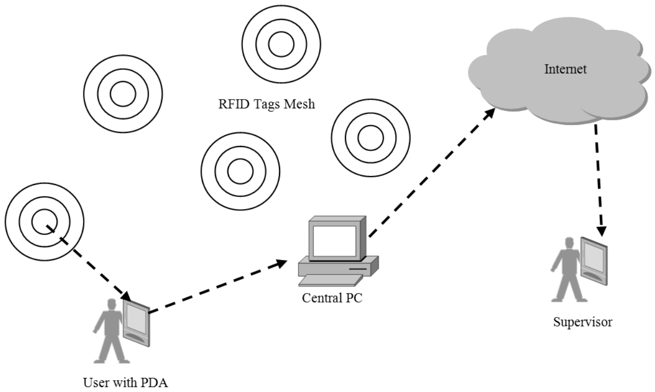

Fig. 1. Basic Monitoring Schema

4. Surveillance Routes Planning Mechanism

For surveillance routes calculation, the system takes into account the time and the minimum distance to be covered. So it is necessary a proper control points grouping and order on each group. The planning mechanism uses Kohonen SOM (Self Organizing Maps) neural networks with the k-means learning algorithm [10] to calculate the optimal routes and assign them to the available security guards.

Neural networks allow the calculus of variable size data collections, and reduce the time and distances to be covered. In addition, the control points can be changed on each calculation, so the surveillance routes are dynamic, avoiding repetitive patterns.

The mechanism starts spreading the control points among the available security guards. Then, the optimal route for each one is calculated using a modified SOM neural network. The modification is done through a FYDPS neural network, changing the neighbourhood function defined in the learning stage of the Kohonen network. The new network has two layers: *IN* and *OUT*. The *IN* layer has two neurons, corresponding the physical control points coordinates. The *OUT* layer has the same number of control points on each route [8][11].

Be $x_i \equiv (x_{i1}, x_{i2}) \quad i = 1, \cdots . N$ the *i* control point coordinates and $n_i \equiv (n_{i1}, n_{i2}) \quad i = 1, \cdots, N$ the *i* neuron coordinates on $\Re^2$, being *N* the number of control points in the route. So, there are two neurons for the *IN* layer and *N* neurons for the OUT layer. The weight actualization formula is defined by the following equation:

$$w_{ki}(t+1) = w_{ki}(t) + \eta(t) g(k,h,t)(x_i(t) - w_{ki}(t)) \qquad (1)$$

Be w_{ki} the weight that connect the *IN* layer *i* neuron with the OUT layer *k* neuron. *t* represents the interaction; $\eta(t)$ the learning rate; and finally, $g(k,h,t)$ the neighbourhood function, which depends on three parameters: the winner neuron, the actual neuron, and the interaction.

A decreasing neighbourhood function is considered with the number of interactions and the winner neuron distance.

$$g(k,h,t) = Exp\left[\left(-\frac{|k-h|}{N/2}\right)\frac{\underset{\substack{i,j\in\{1,\cdots,N\}\\ i\neq j}}{Máx}\{f_{ij}\} - \sqrt{(n_{k1}-n_{h1})^2 + (n_{k2}-n_{h2})^2}}{\underset{\substack{i,j\\ i\neq j}}{Máx}\{f_{ij}\}} - \lambda\frac{|k-h|t}{\beta N}\right] \qquad (2)$$

λ and β are determined empirically. The value of λ is set to 1 by default, and the values of β are set between 5 y 50. *t* is the current interaction. Its value is obtained by means of $\beta N. Exp[x] = e^x$, where *N* is the number of control points. f_{ij} is the distance between two points *i* and *j*. Finally, $Max\{f_{ij}\}$ represents the maximum distance that joins those two points. Each parameter of the neighbourhood function can be 0 or 1. The neighbourhood function radius value must be close to 0 to update just the winner neuron.

To train the neural network, the control points groups are passed to the *IN* layer, so the neurons weights are similar to the control points coordinates. When all the process concludes, there is only one neuron associated to each control point. To determine the optimal route, the *i* neuron is associated with the *i*+1 neuron, from *i*=1, 2, …, *N*, covering all the neurons vector. A last interval is added to complete the route, associating the *N* neuron with the *i* neuron. The total distance is calculated adding the distances between two points. The learning rate depends on the number of interactions, as can be seen on the following equation:

$$\eta(t) = Exp\left[-\sqrt[4]{\frac{t}{\beta N}}\right] \qquad (3)$$

The neurons activation function is the identity. When the learning stage ends, the winner neuron for each point is determined, so each point has only one neuron associated. The optimal route is then calculated following the weights vector. This vector is actually a ring, where the n_1 neuron is the next n_N neuron. Initially considering a high neighbourhood radius, the weights modifications affect the nearest neurons. Reducing the neighbourhood radius, the number of neurons affected decrease, until just the winner neuron is affected. The learning stage is stopped when the distance between two points can not be optimized any more.

The initial number of interactions is $T_1 = \beta N$ in the first stage. When $t = \beta N$, the weights of the possible couple of neurons are changed from the neurons ring obtained. If the distance is optimized, the number of interactions is reduced to continue the learning.

In the Z phase, the total number of interactions is:

$$T_Z = T_{Z-1} - \frac{T_{Z-1}}{Z} \qquad (4)$$

The objective of these phases is to avoid the crossings. Once all interactions are concluded, the distance obtained is analyzed to determine if it is the optimal distance. So, the recoil in the number of interactions is reduced each time, obtaining a maximum number of interactions, although the value is variable. Fig. 2 shows the routes calculated for one and two security guards using the mechanism described before.

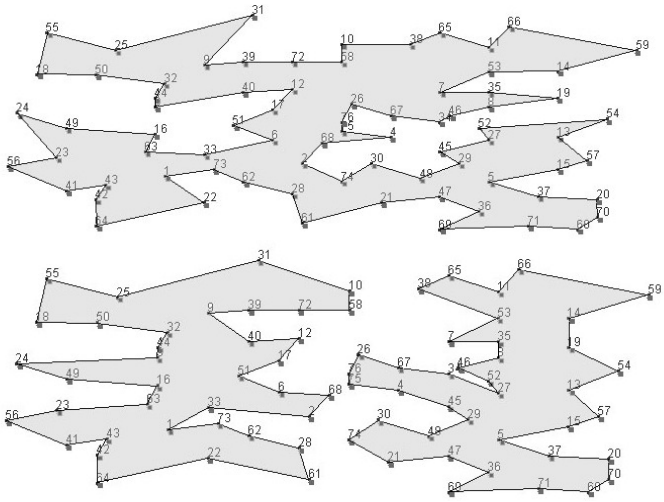

Fig. 2. Planned routes for one (up) and two (down) security guards

5. System Architecture

Multi-agent systems are distributed systems based on cooperation of multiple autonomous agents [3]. Multi-agent architectures comprise plenty errors recovery, with the ability to initialize or end separate agents without the need to restart the entire system [14]. The developed system presented on this paper has these features, so it is possible to initialize multiple services on demand. The agents' behaviour is affected when it is necessary to schedule a new surveillance route. Besides, the system store continuously its status information, so it can be restarted and recover the last backup information.

The analysis and modelling of the system has been done using a combination of the Gaia methodology [15] and the AUML language [2]. Gaia is a simple methodology with an elevated abstraction level for the engineering of multi-agent systems [15]. The models obtained with Gaia can be adapted using AUML to obtain a model of the multi-agent system closer to its implementation [2].

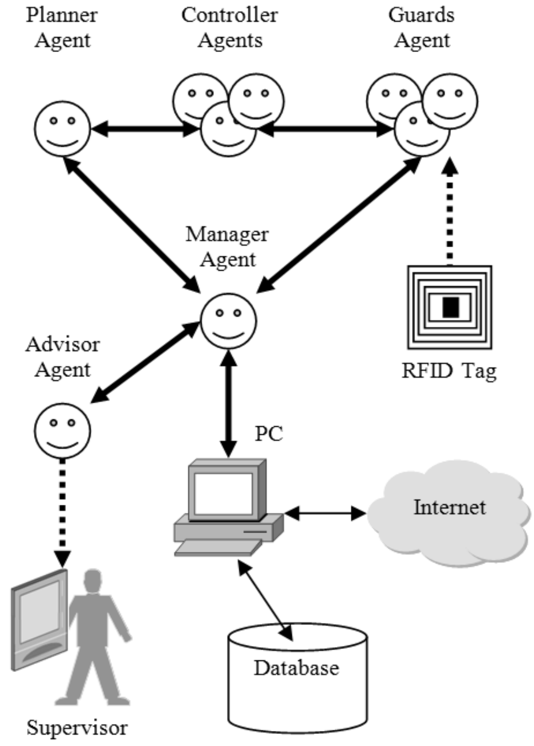

Fig. 3. System Structure

Once defined the system structure, shown on Fig. 3, it is possible to appreciate that the system is composed of five different kinds of agents:

- *Guard Agent.* It is associated to each PDA. Manages the portable RFID readers to get the RFID tags information on every control point. Communicates with *Controller Agents* to check the accomplishment of the assigned surveillance routes, to obtain new routes, and also to send the RFID tags information via Wi-Fi.
- *Manager Agent.* This agent controls all the rest of agents in the system. Manages the connection and disconnection of *Guard Agents* to determine the available security guards available. The information is sent to the *Planner Agent* to generate new surveillance routes. *Manager Agent* also receives incidences (omitted control points, route changes, new security guard connected/disconnected, security guards notifications, etc.) from the *Controller Agents* and *Guard Agents* and, depending its priority,

informs the *Advisor Agent*. *Manager Agent* stores all the system information (incidences, time, data, control points, route status, etc.) into a database. Information can be accessed via Internet.
- *Planner Agent.* Generates automatically the surveillance routes. The routes are sent to the *Manager Agent* to distribute them among the security guards.
- *Controller Agent.* Monitors the security guards activities by means of the control points checked. Once a surveillance route is generated by the *Planner Agent*, the average time to reach each control point is calculated. The *Controller Agent* also handles the associated route incidences and sends them to the *Manager Agent*.
- *Advisor Agent.* Administer the communication with the supervisors (person). Receive from the *Manager Agent* the incidences, and decide if are sent to the supervisor. Incidences can be sent via Wi-Fi, SMS or GPRS. If a communication problem is detected, the incidence is sent back to the *Manager Agent* and stored to be delivered later.

6. Results and Conclusions

The system presented on this paper has been implemented and tested over experimental and controlled scenarios. Simulations have been done to calculate surveillance routes and monitor the accomplishment of each one. The results obtained have shown that it is possible to find out the necessary number of security guards depending on the surveillance routes calculated by the system.

To evaluate the system efficiency, a comparison after and before the prototype implementation was done, defining multiple control points sets and just one security guard. The results of times and distances calculated by the users and the system are shown on Table 1.

Table 1. Time and distance calculated for one security guard and multiple control points sets

Control Points	Users		System	
	Time (min)	Distance (min)	Time (min)	Distance (min)
15	39	1285	28	944
20	64	2124	44	1451
25	76	2535	53	1761
30	79	2617	63	2088
35	96	3191	77	2551

The system provides optimized calculations, so the time and distance are reduced. A complete working day shift can be fixed according the system results, for example, if the route calculated is too long or the time exceeds eight working hours, a new guard must be incorporated.

More simulations were done, defining 3 security guards and 100 control points to determine the level of accuracy compared with the users' predictions. As shown on Table 2, the differences are bigger because the system calculates the optimum route for each security guard and not for the entire control points set.

Table 2. Total time and distance predicted by the users and the system

Control Points	Users		System	
	Time (min)	Distance (min)	Time (min)	Distance (min)
100	357	11,900	212	7,067

Extending these results, Fig. 4 shows the average number of estimated security guards needed to cover an entire area, which consisted on a mesh from 20 to 100 control points, with an increment of 5 control points. The results are clear, for example, for 80 control points, the users estimated 4 security guards, but the system recommended only 3.

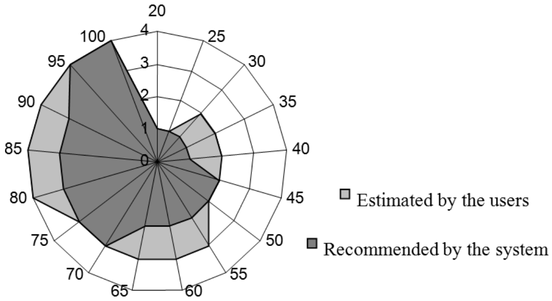

Fig. 4. Average number of estimated security guards

The results obtained so far are positive. It is possible to determine the number of security guards needed to cover an entire area and the loops in the routes, so the human resources are optimized. In addition, the system provides the supervisors relevant information to monitor the workers activities, detecting incidences in the surveillance routes automatically and in real-time.

The use of wireless technologies, such as Wi-Fi, RFID, or GPRS provides an adequate communication infrastructure that the agents can use to obtain information about the context. With this information, the system can adapt services and interact with users according a specific situation in an easy, natural and ubiquitous way to solve some of daily life problems.

The system presented can be easily adapted to other scenarios with similar characteristics, providing a simple but powerful tool to optimize human resources and monitor the staff activities. However, this system is still under development, continuously adding new capabilities and services to have the enough robustness to implement it on real scenarios.

Acknowledgements

This development has been supported by the projects MEC TIN2006-14630-C03-03 and JCYL-2002-05 (SA104A05). Special thanks to Telefónica Móviles and Sokymat for the technology provided.

References

[1] C Angulo and R Tellez, Distributed Intelligence for smart home appliances, Tendencias de la minería de

datos en España, Red Española de Minería de Datos, 2004.

[2] B Bauer and M P Huget, FIPA Modeling: Agent Class Diagrams, 2003.

[3] J M Corchado, J Bajo, Y De Paz, and D I Tapia, Intelligent Environment for Monitoring Alzheimer Patients, Agent Technology for Health Care. Decision Support Systems, Eslevier, Amsterdam, Netherlands, 2007, In press.

[4] T Heikkilä, M Kollingbaum, P Valckenaers and G J Bluemink, An Agent Architecture for Manufacturing Control: manAge. Computers in Industry. No. 46, 2001, pp. 315-331.

[5] Inology, Press note, June 9th, 2005, Available: http://www.controldetiempos.com/sala_de_prensa.htm#absentismo

[6] N Jennings and M Wooldridge, Applications of Intelligent Agents, Queen Mary & Westfield College, University of London, 1998.

[7] H D Jin, K S Leung, M L Wong and Z B Xu, An Efficient Self-Organizing Map Designed by Genetic Algorithms for the Traveling Salesman Problem. IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, vol. 33, No. 6. 2003, pp. 877-888.

[8] J Kleindienst, T Macek, L Seredi and J Sedivy, Vision-enhanced multi-modal interactions in domotic environments, IBM Voice and systems technologies, Czech Republic, 2004.

[9] T Kohonen, Self-Organizing Maps, Springer-Verlag, New York, Inc., 1997.

[10] K S Leung, H D Jin and Z B Xu, An expanding Self-organizing Neural Network for the Traveling Salesman Problem. Neurocomputing, Vol. 62, 2004, pp. 267-292.

[11] Q. Martín, M T Santos and Y De Paz, Operations research: Resolute problems and exercises, Pearson, 2005, pp. 189-190.

[12] P Rigole, T Holvoet and Y Berbers, Using Jini to integrate home automation in a distributed software-system, Computational Sciences Department, Belgium, 2002.

[13] S Garfinkel and B Rosenberg, RFID: Applications, security, and privacy, Addison-Wesley Professional, 2005, pp. 15-36.

[14] M Wooldridge and N R Jennings, Agent Theories, Architectures, and Languages: a Survey, In: Wooldridge and Jennings ed., Intelligent Agents, Springer-Verlag, 1995, pp. 1-22.

[15] M Wooldridge, N R Jennings and D Kinny, The Gaia Methodology for Agent-Oriented Analysis and Design. Journal of Autonomous Agents and Multi-Agent Systems, Vol. 3, 2000, pp. 285-312.

Author Bios

Dante I. Tapia is a PhD. student at the University of Salamanca (Spain) under the supervision of Dr. Juan M. Corchado. He obtained the Engineering in Computer Sciences degree in 2001 and the MSc in Telematics at the University of Colima (Mexico) in 2004. He has been involved in the development of automated systems in the Faculty of Telematics at the University of Colima and deeply collaborating with the Government of the State, where he obtained a scholarship to complete his academic formation. He has also been a co-author of papers published in recognized workshops and symposiums.

Juan F. De Paz is a PhD student at the University of Salamanca (Spain) under the supervision of Dr. Juan M. Corchado. He obtained a Research scholarship from the Science and Education Ministry, and is involved in several R&D projects. He obtained the Engineering in Informatics degree in 2005 at the University of Salamanca (Spain) and de Statistics Diploma in 2007 at the University of Salamanca (Spain). He has also been a co-author of papers published in recognized workshops and symposiums.

Sara Rodríguez a PhD student at the University of Salamanca (Spain) under the supervision of Dr. Juan M. Corchado. She obtained the Engineering in Informatics degree at the University of Salamanca in 2007. She has also been a co-author of papers published in recognized workshops and symposiums.

Javier Bajo received a PhD in Computer Science from the University of Salamanca in 2007. He is an Assistant Professor at the University of Salamanca (Spain). He obtained the Information Technology degree at the University of Valladolid (Spain) in 2001 and Engineering in Computer Sciences degree at the Pontifical University of Salamanca in 2003. He has been a member of the organising and scientific committee of several international symposiums such as CAEPIA, IDEAL, HAIS, etc. and co-author of papers published in recognized journal, workshops and symposiums.

Juan M. Corchado received a PhD. in Computer Science from the University of Salamanca in 1998 and a PhD. in Artificial Intelligence (AI) from the University of Paisley, Glasgow (UK) in 2000. At present he is an Associate Professor, Director of the Intelligent Information System Group (http://bisite.usal.es) and Director of the MSc programs in Computer Science at the University of Salamanca (Spain). He has been a research collaborator with the Plymouth Marine Laboratory (UK) since 1993. He has led several Artificial Intelligence research projects sponsored by the Spanish and European public and private institutions and has supervised seven PhD. students. He is the co-author of over 130 books, book chapters, journal papers, technical reports, etc., most of these present practical and theoretical achievements of hybrid AI and distributed systems. He has been the President of the organising and scientific committee of several international symposiums.