19

Towards Implementing an Intelligent System for Securing and Monitoring using Agents

Faisal Alkhateeb[1], Zain Al-Abdeen Al-Fakhry[2], Eslam Al Maghayreh[3], Ahmad T. Al-Taani[4] and Shadi Aljawarneh[5]
[1,2,3,4]*Yarmouk University, Irbid*
[5]*Al-Isra University, Amman*
Jordan

1. Introduction

Most of organizations such as universities have critical positions and rooms and therefore require to be secured. We mean by "securing" here is detecting and isolation of any problem: temperature, motion, stole attempts, etc. Detection is the phase of announcing a problem while isolation is the phase of determining the type and the location of the problem. Securing a university campus could be achieved based upon a number of security guards making security checking rounds (on foot or by vehicles) inside the university campus and in particular around the critical rooms. Other way is to use security camera-based systems. Nonetheless, such systems need to be monitored on the fly by a person. Additionally, they are inapplicable in some cases such as detecting unseen phenomena (e.g. temperature problems). Finally, cameras should be spread out over critical positions in order to monitor every corner. In this chapter, we present an effective and low security system that operate over a wireless network and based on multi-agents to secure the buildings of a given university.

A wireless sensor network (WSN) is a wireless network consisting of spatially distributed autonomous devices using sensors to cooperatively monitor physical or environmental conditions, such as temperature, sound, vibration, pressure, motion or pollutants, at different locations, and are now used in many industrial and civilian application areas, including industrial process monitoring and control, machine health monitoring, environment and habitat monitoring, healthcare applications, home automation, and traffic control.

The remainder of this chapter is organized as follows: Section 2 provides an overview of some related systems. We introduce in Section 3 the ACCESS architecture and location based services and their applications. Then we introduce our system in Section 4. We present in Section 5 a simulator as well as an implementation of the system.

2. Related systems

In this section, we divided into two main categories: Multi-agent/sensor-based systems and Vision-based systems.

2.1 Multi-agent/sensor-based systems

A multi-agent system (MAS) is a system composed of multiple interacting intelligent agents. Multi-agent systems can be used to solve problems which are difficult or impossible for an individual agent or monolithic system to solve. Multi-agent are open and extensible systems that allow for the deployment of autonomous and proactive software components. For instance, they have been applied to several application areas such as context-aware and infomobility services. The term context refers to any information that can be used to characterize the situation of an entity, where an entity is a person, place, route, train, or any relevant object. So, context-aware services could be defined as those services that deliver "up to the minute" information about a given entity. Such services include location-based services, travel assistance and planning, impaired people mobility assistance. Agent-based approaches are considered as an appropriate solution to address the issue of the overwhelming data traffic in Wireless Sensor Networks (WSNs).

Bus catcher 2 is an agent-based architecture for wireless bus travels assistants (O'Hare et al., 2000; Strahan et al., 2003). The system delivers up to the minute bus network information that is relevant to the transportation and movement of the user.

Another example is EasiShop (Keegan & O'Hare, 2002), which is an agent-based, location-aware, automated ubiquitous commerce (uCommerce) system using mobile agent interaction facilitated by the Bluetooth wireless radio transmission medium.

Gulliver's Genie (O'Grady, n.d.) is an archetypical ubiquitous computing application. Gulliver's Genie uses three basic concepts in its construction, these are: agency, mobility and adaptively. The architecture adopts a Multi-Agent Systems (MAS), where agents manage and maintain a context within which mobile users exist and based upon this context seeks to adapt and personalize content based upon perceived individual user needs.

Another usage of sensors networks and environmental monitoring (Mainwaring et al., 2002). Deploying numerous networked micro-sensors in natural spaces can enable long-term data collection at scales and resolutions that are difficult, if not impossible, to obtain otherwise. Wireless sensor networks have been also used for commercial lighting control (Sandhu et al., 2004), which provides a practical application that can benefit directly from artificial intelligence techniques. This application requires decision making in the face of uncertainty, with needs for system self-configuration and learning. Such a system is particularly well-suited to the evaluation of multi-agent techniques involving distributed learning.

Senor-based systems could be also used to automatically notify emergency centers of a given problem. They are commonly equipped with distributed sensors and used to collect incident severity information. The system then communicates with an emergency dispatcher to assist in determining the appropriate emergency personnel. As an example of such systems is OnStar (Carrigan et al., 2005). It has been designed in 1996 to deliver safety, security and information services using wireless technology and the Global Positioning System (GPS) satellite network. OnStar services include but are not limited to automatic notification of air bag deployment, stolen vehicle location assistance, emergency services, roadside assistance with location, and remote door unlock. It also allows drivers to make and receive voice-activated wireless calls and access a wide range of other information services through a nationwide cellular network. However, such systems have limited usage in the case of University campus since problems that can be occurred in the campus should be treated locally by a particular person. For instance, a temperature problem occurred in a servers' room should be treated by the person from the University responsible for solving that problem. Moreover, such systems cannot be directly applied everywhere because it is required to identify each country' section in order to use the Global Positioning System (GPS) satellite

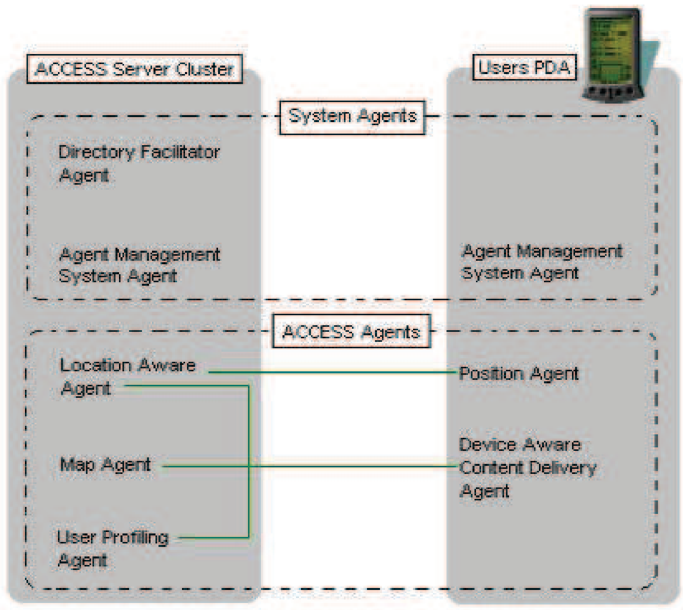


Fig. 1. ACCESS architecture.

network, where a section can be a street, subway, university campus, etc. Additionally, the installation and the maintenance of such systems are too expensive.

2.2 Vision-based systems

The second category of the systems that are closely related to the system presented in this chapter is the vision-based systems. An example of such systems is the vision-based traffic are vision-based traffic accident detection systems (such as traffic Accident Recording and Reporting Systems (ARRS) (Ki, 2007; Ki et al., 2006; Shehata et al., 2008; Wells & Toffin, 2005)). Such systems are either image-based or video-based and used for detecting, recording, and reporting an incident. They consist mainly of a charge coupled device (CCD) camera (located at several places) for monitoring purposes and/or a digital video recorder (DVR) that has recorded all the situations at a given place, and finally an image processing unit that detects images which could be related to an incident. Nonetheless, the usages of such systems are limited since cameras (CCD) and digital video recorders (DVD) cannot be distributed everywhere. Additionally, it is required to have at least one person to monitor the system and alert the security guards in case of a security attack.

3. Research background

In this section, we present the ACCESS architecture since our system is built on top of it as well as Location-Based Services (LBS) that could be used in our system to locate security guards.

3.1 ACCESS architecture

The Agents Channeling (or Conveying) ContExt Sensitive Services (ACCESS) architecture (*cf.* (Strahan et al., 2003)), as shown in Figure 1, consists of two agent categories: system agents and ACCESS agents.

The System Agents manage the platform. The Agent Management System manages the creation and deletion of agents and provides a white pages directory service for agents that reside on that platform. The Directory Facilitator Agent provides a yellow pages service for agents, it has information about agents and the services they provide.

The ACCESS Agents consists of generic agents responsible for channeling context sensitive services. The Device Aware Content Delivery Agent manages the local cache and the user interface to ensure real time content delivery to the user. The Position Agent determines the users physical position and informing specified agents of the user's location and direction. The Location Aware Agent receives location information from the position agent. It also acts as a hotspot manager, where a hotspot is an area of physical space limited by specific boundaries. It can inform specified agents of when the user enters or leaves a hotspot and provides functionality to allow agents register and delete hotspots. The User Profiling Agent tries to determine what type of user it believes is using the system. It monitors the user's location preferences, available hardware and services used. The Map Agent is responsible for dynamically generating maps merged with service specific overlays.

3.2 Overview of location based services

As a result of the huge numbers of mobile phone users, it becomes possible to use the mobile in specifying the location of the mobile users. One method that can be used to accomplish this task is the use of location based services (LBSs). A location-based service (LBS) is an information service that can be accessed using mobile devices through the mobile network and utilizes the ability to make use of the geographical position of the mobile device (Adams et al., 2003; Steiniger et al., 2006; Vijayalakshmi & Kannan, 2009).
There are a number of services included in LBSs such as specifing other people positions, other resources, and the position of the user itself, etc. The primary service is obtaining the position of the user itself in order to use a given service such as finding the nearest restaurant.
There are different types of location based services (Bellavista et al., 2008; Chen et al., 2002; Rao & Minakakis, 2003; Schiller & Voisard, 2004):
Pull services: this type of service is initiated by the client himself by requesting the service from LBS and giving the permission to the LBS to know his/her location. One example of a Pull service is traffic information requested by sending an SMS to a given number such as 1000 that is specialized in this service. The service provider needs to know the location of the client to provide him with information that fulfills the request of the client.
Push services: this type of service is initiated by the service provider as a result of previously getting the authority from the client to receive the requested information from the service provider. For example, a client can register in a traffic service. Every morning when he/she is going to work at 7:30AM, the service provider supplies her/him with information regarding traffic movement at that time depending at her/his current location. Consequently, if there are a huge number of vehicles at a given street it will provide her/him with possible alternatives that she/he can go on so that she/he can arrive at her/his work on time.
Tracking services: this type of services allows someone to request the location of another one. In such type of services, the person whose location is required has to permit the first one to follow him. A client can press a number and send an SMS message to specify a given person location, if the person replay, the client will receive his location via SMS.
Emergency Services: such kind of services provide an automatic or manual call to civil defense in case of an accident or risk. This service should be provided to all mobile clients registered in this mobile company. In USA when you dial 911 from your mobile this service will allow the emergency personnel to specify the location of the caller directly.

4. Campus-agents security system

In this section, we present the system requirements, its architecture and a typical scenario of the Campus-agents security system.

4.1 System requirements

The main components of the system are:

- Wireless sensors and receivers: the wireless sensors will be distributed in critical rooms to detect and to alert the system in case of a security attack occurred.
- Database: the database contains information about sensor flags and locations, security guards, critical campus rooms in which there are sensors.
- PDAs and/or mobiles: security guards will be provided with PDAs or mobiles[1].

4.2 Campus-agents service architecture

Due to the fact that the ACCESS architecture is generic, application specific Service Agents, which use the ACCESS architecture could be added to the system. Therefore to fit the requirements of our system, we extend the architecture as shown Figure 2, which we have proposed in (Alkhateeb et al., 2010). The Service Agents for this application are:

- Alarm Management Agent: its function is to receive the signal from the wireless sensor and determine the code of that sensor and get the building code from the sensor table and send it to the Building Position Agent, it will also get the floor number, the room number and security degree and send them to the Security Guard Call Agent.
- Building Position Agent: its function is to receive the building code from the Alarm Management Agent and will get the building name and position from the building table and send them to the Security Guard Call Agent.
- Security Guard Call Agent: it receives the information from the Alarm Management Agent and the Building Position Agent and will find the shortest distance between the building in which the alarm activated and the nearest free guard (by checking the security guard position table), then it will send the building position to the Map Agent to highlight the building on the map displayed in that guard's PDA and send a text message to the same PDA telling the guard to go to that building and to the room where the sensor is located.

4.3 Typical scenario of the system

The principal use of the Campus-agents will be detecting security attacks. Basically, the system will work as follows:

If one of the wireless sensors detects an attack occurring at some room in one of the university buildings, then it sends a signal to the system. The Alarm Management Agent receives the signal and determines the room code by searching the sensor table of the database based on the identifier of the sensor sending the signal.

The Building Position Agent receives the code from the Alarm management Agent and will get the building name and position from the building table and send them to the Security Guard Call agent.

Then the Security Guard Call agent receives the information from the Alarm Management Agent and the Building Position Agent and will find the shortest distance between the building in which the alarm activated and the location of security guards. Also it identifies the nearest free guard by checking the security guard position table, pass the building position to the Map Agent to highlight the building on the map displayed in that guard's PDA, and send a text message to the same PDA telling the guard to go to that building and to the room where the sensor is located. One usage of the Position agents and Location agents is to check whether a given security guard is out the zone of the University campus or not. This could

[1] Note that it is assumed that everyone posses a mobile and hence this requirement could be ignored.

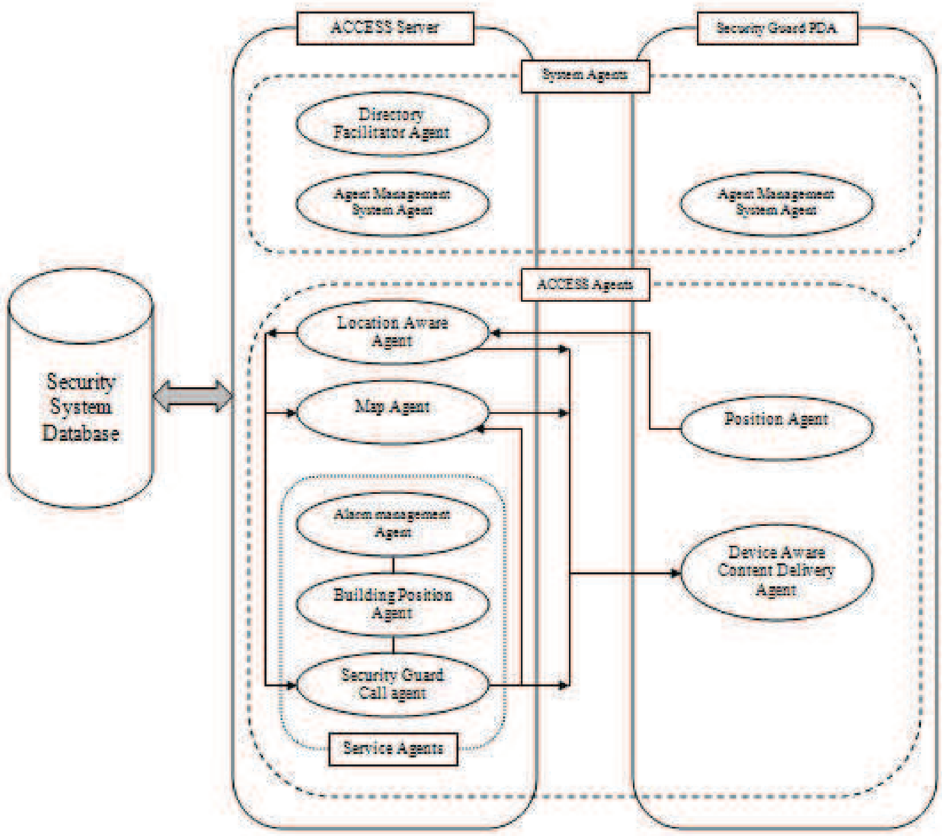


Fig. 2. Campus Service Architecture.

be achieved by using location-based services discussed earlier. Based on that information, the system will allocate other security guards to fill his position.

5. System simulation and implementation

This section introduces the components that we have used in our implementation. For more details, the reader could refer to [16].

5.1 Agent factory agent programming language 2 (AFAPL2)

ACCESS was originally built using Agent Factory Micro Edition (AFME) (C. Muldoon, 2006), which is a minimized footprint agent platform developed to enable the creation of intentional agents for mobile devices. Due to the fact that the NetBeans IDE does not compile AFME, we used the AFAPL2, which is an extended version of the original Agent Factory Agent Programming Language (AFAPL) (cf. (Collier, 2007)).

5.2 Databases

The database we constructed contains the following tables:

- Attack table: this table is used to make reports on the attacks detected by the sensors and dealt with by the guards during a specified time period.
- Building table: this table is used to store the buildings' name and co-ordinations.
- Securityguard table: this table is used to store the security guards information.
- Sensors table: it is used to store the sensors information.

Fig. 3. System simulation using laptops.

- Sgposition table: this table is used to store the security guards positions.

5.3 System simulation

Without loss of generality, we have used laptops to simulate the system. This does not provide any restriction to the usage of our system. That is, the wireless sensors could be integrated to the system by first adding their codes and/or their IP addresses to the sensor table mentioned above. Then, they will be connected to the system using a number of access points deployed in the system. Types of sensors include but not limited to glass break detectors, door and window open detectors, motion detectors.
In the simulation, we used two laptops: one to represent the wireless sensor and the other is the server both equipped with built-in wireless network card (WiFi). We used a Wireless router as an access point. We gave them fixed IP addresses.
The classes "java.net.Socket" and "java.net.ServerSocket" are used to implement sensor communications which use the TCP protocol (see Figure 3).

5.4 System work flow

At the beginning of the system execution the start window will be displayed and the user has to click the start button to display the rest of the windows.
From the sensors window the user will select a sensor (which is a simulation of the sensors control panel) and click it to set the alarm of that sensor on.
The alarm management agent, which its task to monitor the wireless sensors signals, will detect the new alarm signal and then it will signal the building position agent to act. After receiving the signal from the Alarm Management Agent, the building position agent will call some classes to link to the sensors table to get the building name, floor number and room number where the sensor is deployed. These will also create the calling message that will be displayed on the security guard PDA. The building position agent will also call another class to inquire the building table to get the co-ordinations of the building, and then it will send them to the security guard call agent.
The security guard call agent contains several classes that are used to inquiry the Sgposition table and calculate the distance between each free security guard and the building to select

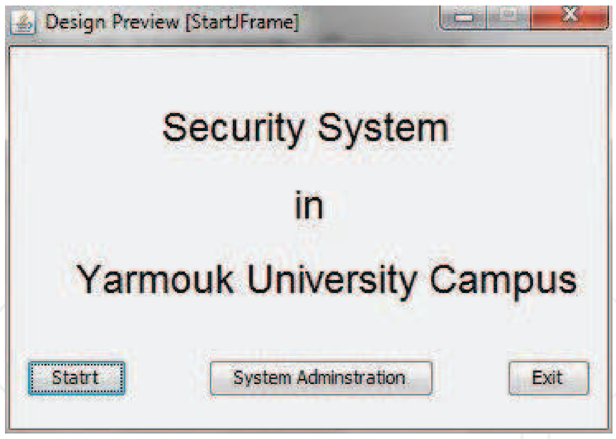


Fig. 4. System start menu.

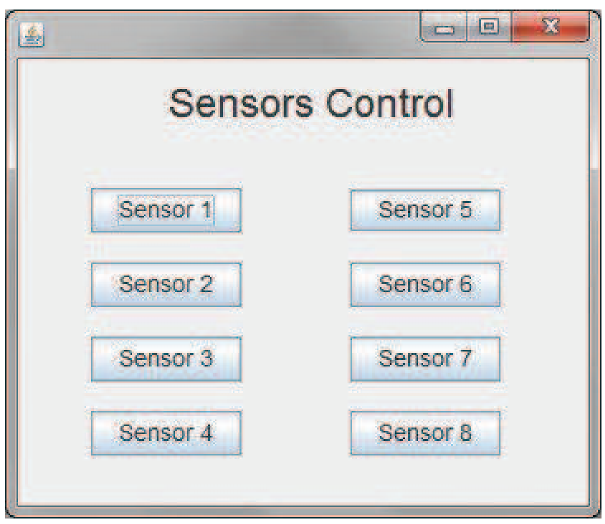


Fig. 5. Sensors control menu.

the nearest available security guard to the building. Then it will send two signals, one to the map agent and the other one to the Device aware content delivery agent.
The map agent in tern will select the map that will be displayed in the security guard PDA, and then it will send a signal to the Device aware content delivery agent.
The Device aware content delivery agent will receive the map address from the map agent and display the map and a message on the security guard PDA as shown below.

6. Conclusion and future work

We have proposed in this chapter a novel system that can be employed to build an effective and low cost security system made by wireless sensor network and multi agents. The intelligent agents (with the aid of sensors) are responsible for detecting security attacks and alerting security guards. Consequently, there is no need for a human to continuously monitor the rooms and the buildings of the University campus. Additionally, the system could be used to secure any organization containing critical locations.
Although we have used laptops in the implementation to simulate the functions sensors, real sensors could be integrated seamlessly to the system.
As a future work, we plan to investigate the usage of cameras in conjunction of this system and therefore use the image processing techniques to extract security attacks features. In

Fig. 6. A message and a map displayed in the PDA.

this direction, in case of personal attacks, location-based services could be used to track the attacker as discussed in (Al-Fakhry, 2010).

7. References

Adams, P. M., Ashwell, G. W. B. & Baxter, R. (2003). Location-based services — an overview of the standards, *BT Technology Journal* 21(1): 34–43.

Al-Fakhry, Z. (2010). Towards implementing intelligent security system.

Alkhateeb, F., Al Maghayreh, E. & Aljawarneh, S. (2010). A multi agent-based system for securing university campus: Design and architecture, *Proceedings of the International Conference on Intelligent Systems, Modelling and Simulation (ISMS'10)*, IEEE, Liverpool, England, pp. 75–79. URL: *http://doi.ieeecomputersociety.org/10.1109/ISMS.2010.70*.

Bellavista, P., Küpper, A. & Helal, S. (2008). Location-based services: Back to the future, *IEEE Pervasive Computing* 7(2): 85–89.

C. Muldoon, G. M. P. O'hare, R. C. M. J. O. (2006). Agent factory micro edition: A framework for ambient applications, *Proceedings of Intelligent Agents in Computing Systems Workshop (held in Conjunction with International Conference on Computational Science (ICCS)) Reading, UK. Lecture Notes in Computer Science (LNCS)*, Springer-Verlag, pp. 727–734.

Carrigan, R., Milton, R. & Morrow, D. (2005). ADVANCED AUTOMATIC CRASH NOTIFICATION (AACN), *Computerworld honors case study*. URL: *www.onstar.com/* URL: *www.cwhonors.org/archives/case_studies/onstar.pdf*.

Chen, Y., Chen, X., Ding, X., Rao, F. & Liu, D. (2002). Bluelocator: Enabling enterprise location-based services, *Mobile Data Management, IEEE International Conference on* 0: 167.

Collier, R. (2007). Debugging agents in agent factory. programming multi-agent systems, *Lecture Notes in Computer Science (LNCS)*, Springer- Berlin/Heidelberg, Springer Berlin/Heidelberg, pp. 229–248.

Keegan, S. & O'Hare, G. (2002). Easishop: Context sensitive shopping for the mobile user through mobile agent technology, *Proceedings of PIMRC 2002 13th IEEE International Symposium on Personal Indoor and Mobile Radio Communications, IEEE Press*.

Ki, Y.-K. (2007). Accident Detection System using Image Processing and MDR, *International Journal of Computer Science and Network Security IJCSNS* 7(3): 35–39. URL: *http://paper.ijcsns.org/07_book/200703/20070306.pdf*.

Ki, Y.-K., Kim, J.-W. & Baik, D.-K. (2006). A Traffic Accident Detection Model using Metadata Registry, *Software Engineering Research, Management and Applications, ACIS International Conference on Software Engineering Research, Management and Applications* pp. 255–259.

Mainwaring, A., Polastre, J., Szewczyk, R., Culler, D. & Anderson, J. (2002). Wireless sensor networks for habitat monitoring, *Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications, SESSION: Applications and OS*, Atlanta, Georgia, USA, pp. 88–97.

O'Grady, M.J., O. G. (n.d.). Gulliver's genie: Agency, mobility & adaptivity, *Computers & Graphics, Special Issue on Pervasive Computing and Ambient Intelligence - Mobility, Ubiquity and Wearables Get Together* 28(4, year = 2004,

O'Hare, G., O'Hare, P. & Lowen, T. (2000). Far and a way: Context sensitive service delivery through mobile lightweight PDA hosted agents, *Proceedings of 15th International Florida Artificial Intelligence (FLAIRS) Conference*, AAAI Press, Pensacola, Florida.

Rao, B. & Minakakis, L. (2003). Evolution of mobile location-based services, *Commun. ACM* 46(12): 61–65.

Sandhu, J. S., Agogino, A. M. & Agogino, A. K. (2004). Wireless sensor networks for commercial lighting control: Decision making with multi-agent systems, *Berkeley, ca 94720, moffett field, ca 94035*. URL: *best.berkeley.edu/ aagogino/papers/AAAI_04_SN_J_Sandhu_ final.pdf*.

Schiller, J. H. & Voisard, A. (eds) (2004). *Location-Based Services*, Morgan Kaufmann.

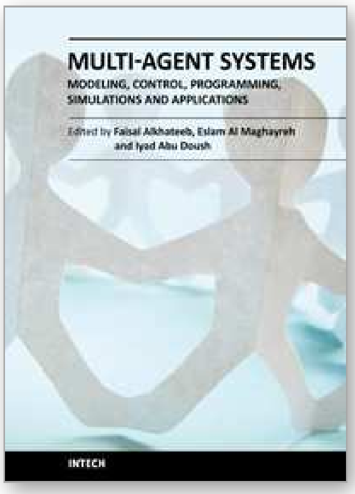
Shehata, M. S., Cai, J., Badawy, W. M., Burr, T. W., Pervez, M. S., Johannesson, R. & Radmanesh, A. (2008). Video-based automatic incident detection for smart roads: The outdoor environmental challenges regarding false alarms, *IEEE Transactions on Intelligent Transportation Systems* 9(2): 349–360.

Steiniger, S., Neun, M. & Edwardes, A. (2006). Foundations of location based services, *Cartouche - lecture notes on lbs, v. 1.0*. URL: *http://www.geo.unizh.ch/publications/cartouche/ lbs_lecturenotes_steinigeretal2006.pdf*.

Strahan, R., Muldoon, C., O'Hare, G., O'hare, G. M. P., Bertolotto, M. & Collier, R. W. (2003). An agent-based architecture for wireless bus travel assistants, *Proceedings The Second International Workshop on Wireless Information Systems (WIS 2003)*.

Vijayalakshmi, M. & Kannan, A. (2009). Proactive location-based context aware services using agents, *International Journal of Mobile Communications* 7(2): 232–252.

Wells, T. & Toffin, E. (2005). Video-based automatic incident detection on san-mateo bridge in the san francisco bay area, *Proceedings of 12th World Congress on ITS*, San Francisco.



Multi-Agent Systems - Modeling, Control, Programming, Simulations and Applications
Edited by Dr. Faisal Alkhateeb

ISBN 978-953-307-174-9
Hard cover, 522 pages
Publisher InTech
Published online 01, April, 2011
Published in print edition April, 2011

A multi-agent system (MAS) is a system composed of multiple interacting intelligent agents. Multi-agent systems can be used to solve problems which are difficult or impossible for an individual agent or monolithic system to solve. Agent systems are open and extensible systems that allow for the deployment of autonomous and proactive software components. Multi-agent systems have been brought up and used in several application domains.

InTech Europe
University Campus STeP Ri
Slavka Krautzeka 83/A
51000 Rijeka, Croatia
Phone: +385 (51) 770 447
Fax: +385 (51) 686 166
www.intechopen.com

InTech China
Unit 405, Office Block, Hotel Equatorial Shanghai
No.65, Yan An Road (West), Shanghai, 200040, China
中国上海市延安西路65号上海国际贵都大饭店办公楼405单元
Phone: +86-21-62489820
Fax: +86-21-62489821