Int. J. of Sustainable Water and Environmental Systems
Volume 5, No. 2 (2013) 59-65

Agents and Sensors System for Monitoring Sandstorms

Elhadi M. Shakshuki [a]*, Tarek R. Sheltami [b]

[a] *Acadia University, Wolfville, Nova Scotia, Canada, B4P 2R6*
[b] *King Fahd University of Petroleum & Minerals, Dhahran, Saudi Arabia, 31261*

Abstract

Saudi Arabia highways, especially in the eastern region, are subject to several sandstorms and sand dunes that yield to reduced visibility, unsafe roads and sometimes it is life threatening. This makes wireless sensor networks bring very promising, desired and effective natural solution to provide safe roads. This paper presents agent-based system architecture approach for data dissemination in a wireless sensor network (WSN) for sandstorm monitoring. This system architecture consists of three layers of agents, including the interface, routing and data collection layer. The interface agent responsibility is to interact with the users to fulfill their interests. The routing agent generates the optimized query plan to the cluster agent. At the routing layer, the routing agents perform data dissemination and efficient in-network processing with the other agents in the same layer; it also captures the required data through the data collection layer that has direct access to sensor nodes. This paper provides the agents' architecture, design and implementations that enable them to communicate and work together to disseminate and gather data in WSNs.

Keywords: *Agents, data dissemination, query optimization, sensor networks*.

1. Introduction

The highways in the kingdom of Saudi Arabia are subject to several sandstorms and sand dunes, especially in the eastern region. As a result, the visibility will be reduced which makes the roads unsafe and sometimes life threatening. Wireless sensor networks combined with intelligent software agents provide a natural solution to make such roads under control and safe.

Because of recent advances in electronics and Micro-Electro-Mechanical Systems (MEMS), it is feasible to integrate computing and storage resources with the typical monitoring sensors. These smart sensors can be connected using a serial cable, Ethernet cable, radio, acoustics, or any other physical medium of connectivity [1]. Sensor networks offer real time continuous monitoring, which is precise, accurate, reliable, as well as cost effective solution [2].

Wireless sensor network enables pervasive, ubiquitous and seamless communication with the physical world. Moreover, sensor networks promise efficient solutions for various applications such as military, security, automation, energy resources, and environmental monitoring [22]. However, there exist certain limiting factors such as: fault tolerance, scalability, cost, hardware, topology change, and power consumption [7]. This makes it a necessity to deal with these issues and develop several techniques and mechanisms to provide satisfactory solutions to them.

The main aim of this paper is to investigate the architecture and design of smart sensor networks for sandstorm monitoring. Although significant advances have been made in the areas of remote sensing, much of these technologies remain untouched by transportation ministry. Sandstorms may easily impact climate and air temperature [4]. Dust may affect soil formation, create geomorphologic formations in dry lands, and remove cover from desert surfaces. At the same time dust can transmit pathogens harmful to humans, impair respiratory function, cause air pollution, and disrupt communications and transport. Most active dust storm sources can occur in truly remote areas where there is little or no human activity though many sources are associated with areas where human impacts are well documented [3].

The proposed remote sandstorm sensing system will address the requirements and incorporate technologies for transmitting sand movement data for real-time access from any desired location. The sandstorm monitoring application will be based on heterogeneous environment that will consist of several independent sub-networks. There is a need for an agent-based middleware framework that will provide the

* Corresponding author. Tel.: +19025851524
Fax: +19025851067; E-mail: elhadi.shakshuki@acadiau.ca

DOI: 10.5383/swes.05.01.009

abstraction for the different technologies to be used. One promising solution is to use an agent-based approach [5], where the agent has the ability to communicate with all the sensors in the environment, create clusters and sub-networks, learn about the status of all sensors, analyze the decisions, and act autonomously.

The proposed agent system architecture has several components. One of its main components is the learning capability, which can be implemented by a Genetic Algorithm technique. The agent acts as a mediator between the administrator and the sensor network environment. This agent provides real-time solutions for energy-efficient query processing and data dissemination. The agent-based architecture will use a database approach where the users will be unaware of the implementation details. In other words, users would be able to query the network without worrying about the energy efficient query processing and routing techniques.

The architecture consists of three layers of agents with different types of functionalities, including the interface layer, routing layer and data collection layer. At the interface layer, the interface agents interact with the users to fulfill their interests. The routing agent generates the optimized query plan to the cluster agent. At the routing layer, the routing agents perform data dissemination and efficient in-network processing with the other agents in the same layer; it also captures the required data through the data collection layer that has direct access to sensor nodes. Using this layered architecture, the application developers could easily build web-based or standard applications for the given sensors, without going into the hardware, routing or sensor details.

The rest of this paper is organized as follows: Section 2 introduces wireless sensor network, sensor motes, database approach and intelligent agents. Section 3 discusses some related work in data dissemination and aggregation. Section 4 provides the agent-based WSN environment and proposed system architecture; followed by the architecture of the routing agent in Section 5. A sample prototype implementation is given in Section 6. Finally, Section 6 presents the conclusion.

2. System Requirements

All fast, reliable, and flexible agent systems provide quick response, reliability and flexibility for tomorrow's Wireless Sensor Network (WSN) applications [7]. Unlike regular programs, agents with mobility feature can execute special migration instructions that move both code and state from one node to another, thereby allowing them to control their execution context. By having several agents collecting the desired sensed data in parallel, larger regions of the WSN can be covered with less time. In addition, agents provide a higher degree of reliability since they can be programmed to intelligently adapt to unexpected situations such as changes in the network topology. Moreover, agents can be dynamically injected into a pre-existing network, meaning the network can be reprogrammed and multiple programs can run simultaneously. This section provides background information about wireless sensor networks, database approach and intelligent agents.

2.1. Wireless Sensor Networks

A wireless sensor network comprises numerous sensor devices, commonly known as motes, which can contain several sensors to monitor the physical entities such as temperature, light, motion, metallic objects, humidity, etc. [6]. The sensor motes are spatially scattered over a large area. Since the deployment environment could be harsh, hostile, inaccessible, and remote, the network must possess easy deployment and maintenance free strategies. The unique combination of mote characteristics, deployment environment, and diverse applications has generated a lot of interest in the research community [7].

The pervasive nature of static and mobile sensors scattered throughout the environment enables multi-resolution capture of environmental information that serve as the basis for knowledge acquisition, management and decision-makings. Information processing for sensor networks enables new capabilities to perceive and operate autonomously, and adaptively react to their environments.

Because of lowering upfront costs and reducing operating expenses, wireless sensor network provides improved reliability, increased installation flexibility, and scalability to sensor networks. It also makes numerous monitoring applications feasible that were previously not possible due to remote and hazardous environments.

2.2. Sensor Motes

As the examples of commercially available motes would be beneficial in appreciating the architecture and design of WSN, this work utilizes the commonly used motes of mica [11]. A Crossbow mica2 mote is shown in Fig. 1. In mica platform, mote, sensors, and programming board are designed on three distinct boards. Motes (e.g., Crossbow's mica2, mica2dot, and micaz) are programmed by programming boards MIB510 and MIB600 that use RS-232 serial interface and Ethernet gateway respectively. The sensor data is obtained by attaching a sensor board, MIB300 or MIB310, to the mote.

Fig. 1. Sensor node

2.3. Database Approach

Traditionally, sensor networks consist of numerous sensors that produce continuous data streams. Due to the inherent nature of sensor networks, the database approach is preferred to execute different queries for the sensor data streams. Moreover, the database approach provides an abstraction for the data storage, allocation, and distribution. For a user, all the data appears to be local and all the queries could be performed without writing any energy efficient algorithm. A few database approaches for sensor networks are described. Madden et al. [12] proposed a Tiny Aggregation (TAG) Service for data aggregation. This approach provides a declarative interface for the data collection and in-network aggregation. It performs intelligent distributions of queries and executes them efficiently by optimum utilization of constrained recourse of the network. At each node, irrelevant data is discarded; the relevant data are merged using aggregate functions.

2.4. Intelligent Software Agents

An intelligent software agent is a computational system having goals, sensors and effectors, and can decide autonomously which action to take and when [9]. An agent is anything that can be viewed as perceiving its environment through sensors and acting upon that environment through effectors. A software agent as a computer program resides in a system, network or a device and is activated upon some particular event or change in the environment [8]. Software agents can also communicate with other software agents in order to meet their individual or common goals. A software agent acts based on the preferences of the user. In other words, the software agent has the ability to analyze the behaviour of the user. For example, if there is a task to “find the picture of the red fishes,” the software agent acts accordingly and searches for it on behalf of the user. Similarly, if it is given the task of “regular check of machine for viruses” the software agent will regularly search for a virus and will generate alerts in case of an intrusion by a new virus [10]. Thus, a software agent makes life easy and efficient for users by assisting them to find the desired solutions.

Intelligent software agents are designed to fulfill the requirements of the domain in which they are needed. Software agents have special tasks to perform. Depending upon the tasks, the characteristics (e.g., autonomy, reactive, communicative, goal oriented, temporal continuity, proactive) are added to the software agent. It is not necessary for a given software agent to possess all the characteristics. Some are autonomous, some are entertaining, some provide information, and some possess learning capability.

3. Data Dissemination and Aggregation

The application of WSN is growing fast and has made a significant contribution to the problem of providing different strategies for efficient data dissemination and data aggregation. For example, Harshavardhan and Kishnendu [13] have proposed a location-aided flooding protocol to achieve energy efficient data dissemination in WSN. This protocol divides the network into virtual grids, where each grid contains either gateway nodes or internal nodes. Gateway nodes are responsible for transmitting data across the grids whereas the internal nodes transmit data within the grid. This proposed protocol focused on reducing redundant transmission to achieve energy efficient data dissemination.

A distributed service protocol for wireless sensor network for query processing is proposed in [14] to target specific data tracking. In this work, the proposed distributed services and network protocol aimed at addressing problems of mobility, dispersion, dynamic reconfiguration and limited power supply. The services provided to the sensor network enable the network nodes to collaborate with each other for accomplishing various coordinated tasks and network functionalities. Further, the services enable the nodes to be aware of the capabilities and functionalities of their neighbour nodes.

Heinzelman et al. [15] described the LEACH protocol, which is a hierarchical self-organized cluster-based approach for monitoring applications. The data collection area is randomly divided into several clusters, where the number of clusters is pre-determined. Based on time division multiple access, the sensor nodes transmit data to the cluster heads, which aggregate and transmit the data to the base station. Bandyopadhyay and Coyle [16] described a multi-level hierarchical clustering algorithm, where the parameters for minimum energy consumption are obtained using stochastic geometry.

Other researchers such as Jin et al. [17] have used genetic algorithms for energy optimization in wireless sensor networks. In their work, GA allowed the formation of a number of pre-defined independent clusters which helped in reducing the total minimum communication distance. Their results showed that the number of cluster-heads is about 10% of the total number of nodes. The pre-defined cluster formation also decreased the communication distance by 80% as compared with the distance of direct transmission. Ferentionos et al. [17,18] extended the attempts proposed by Jin et al. [17] by improving the genetic algorithm fitness function. The focus of their work is based on the optimization properties of genetic algorithm for monitoring sandstorms.

This paper focuses on viewing data dissemination as a distributed problem that can be solved by collaborative agents, for which several agents work together with specialized functionality in a layered architecture.

The operations of the network are divided into three layers. Specific task oriented agents control each layer. Agents in each layer achieve their assigned tasks by collaborating with each other and with other agents in other layers and transmit the processed data to the next layer. The system architecture includes three layers: interface layer, routing layer and data collection layer. The interface layer provides flexible mediation services between users and the WSN environment. The routing layer flexibly links between other agents of the system by facilitating collaboration and generating the optimized query plan. The routing layer captures the required data and performs data dissemination and efficient in-network processing. The data collection layer acts as data provider for the routing layer.

4. Agent-based System Architecture

A brief description of the environment is provided in this section, with a special focus on the routing agent architecture. The architecture of the proposed agent-based system is designed to help users to monitor and receive information about sandstorms from distributed sensor motes. Each agent is autonomous, cooperative, coordinated, intelligent, rational and able to communicate with other agents to fulfill the users' needs. Fig. 2 shows our agent-based environment for sensor network.

This environment consists of four entities including: user, user application, software agents, and sensor nodes. The user is the administrator who manages the network. Application is the gateway for the user to submit queries and receive results. Agents are the intelligent entities that are able to respond to the user needs and relieve the user from being the controller. Sensor nodes are physical entities that are able to read different phenomenon from the environment.

Routing agent resides at the base station performs computation intense intelligent operations, such as genetic algorithm operations. However, as sensor nodes have limited energy supply, a sensor node is equipped with a light-weighted data collection agent. Although all the agents have the same structure [19], the functionality of each of them is adapted according to the requirements. An interface agent is responsible to send queries and receive the results.

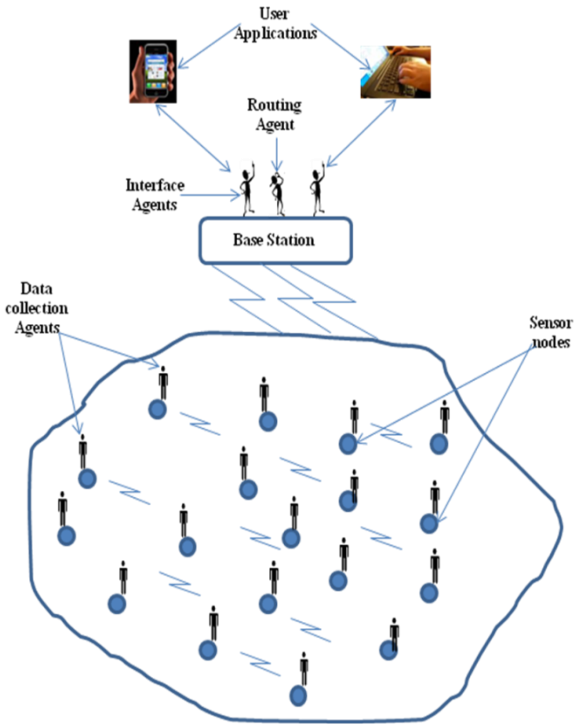

Fig. 2. The environment

4.1. Query Processing

A desktop can be used as one of the standard interfaces such as, a tablet PC, or a personal digital assistant (PDA), a user initiates a query to the sensor network application. The application sends the query to the interface agent of the base station. The interface agent in turn sends the query to the routing agent's communication component. When the routing agent receives the query, it forwards it to the sensor network as packets. Since the sensor network application will consist of various types of sensors and communication technologies, the communication component will act as a bridge between different forms of sensor subnets. The communication component will forward the packets in formats that are applicable for the corresponding connected sensor subnets. For instance, some areas of the sand monitoring will include fibre optics sensor subnets; whereas other areas will be monitored by the remote communication and conventional sensor subnets.

The network is divided into several subnets based on the communication technology. Moreover, for energy-efficient routing the subnet is divided into several clusters, where a cluster head manages each cluster. The cluster head is responsible to collect sensor data from the cluster members. Then, the sensor data is aggregated and transmitted to the base station. Although the clusters are self-organized, the base station can assist in the formation of clusters and the creation of transmission schedules.

When the routing agent sends packets to the sensor subnets, the agent updates its knowledge with the user queries and interests. At the sensor nodes, the data collection agents are responsible for retrieving the results. The data collection agents send their results to their cluster head agents. The cluster head agents then aggregate the query results and send their aggregated results to the routing agent.

5. Routing Agent Architecture

The proposed routing agent architecture consists of several components to perform real-time energy efficient routing decisions. The routing agent has four executable components and a knowledge base repository, as shown in Fig. 3. The routing agent executable components are as follows: communication, problem solver, assignment, scheduler, learning, and knowledge update.

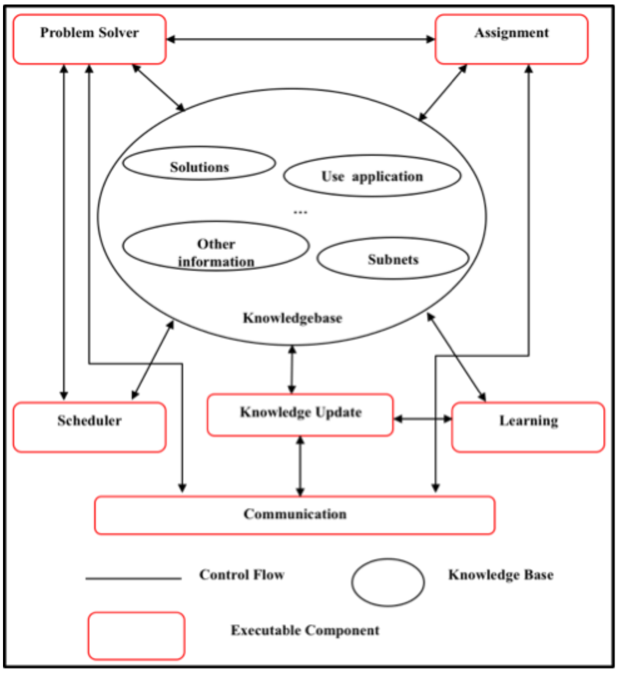

Fig. 3. Routing agent architecture

The *communication* component allows the agent to exchange messages with the other elements of the environment. The communication component creates packets that are transmitted to the desired sensor nodes. The communication components of the data collection (at the sensor nodes) receive the packets broadcasted by the routing agent at the base station. The data collection agents record the appropriate network and cluster information in their knowledge base repositories.

The *problem solver* component is a decision maker that analyzes the contents of the knowledge base to choose an optimum routing plan for an optimum number of future transmissions. Moreover, the problem solver performs the functions of damage detection, modal analysis, and non-destructive system evaluation. In other words, as the routing agent performs cross layer optimization, the optimization includes the data dissemination, as well as estimation of data uncertainty, reliability, and accuracy. The problem solver component sends the decision to the assignment component.

- The *assignment* component assigns or delegates the responsibilities to data collection agents at sensor nodes. For instance, in case of warning or alerts, the assignment operator sends the alerts messages to the suitable agents or entities. The assignment component sends and receives all messages through the communication component.
- The *scheduler* component provides routing agent with a time agenda to start or stop certain activities such as monitoring and aggregating data.
- The routing agent's *learning* component uses genetic algorithm to provide an optimum network configuration for an optimum number of future transmissions. First, it creates an initial population with a large number of clusters, where the available nodes and start energies are obtained from the knowledge base repository. Then, several generations are created to produce an optimum chromosome. The optimum solution provides the complete network details: the number of cluster heads, the members of each cluster head, and the number of transmissions for this configuration.
- The *knowledge base* component stores the optimum solution generated by the learning process. In addition to optimum solutions, the knowledge base repository contains all the information about the network conditions, user applications, types of sensor subnets, sand temperature, wind speed and direction, and other related sandstorm details.
- The data collection agents transmit packets according to the given plan for a given number of transmissions. Once the current plan is completed, the problem solver component chooses another plan from the knowledge base and the network is reconfigured (or assigned) according to the new plan. A round is defined as a time interval in which a new optimum plan is generated and the optimum number of transmissions is transmitted according to the given plan. The time interval between successive transmissions is an application dependent parameter and can be adjusted by the user.

5. An Application Example

To demonstrate our proposed system, a sensor network application is designed and implemented, using 18 mica2 motes. The mica2 motes are attached with MIB310 sensor boards. The application monitors temperature, light, and mobility in two buildings: Carnegie Hall (CAR) and Huggins Science Hall (HSH) at Acadia University, Wolfville, Nova Scotia, Canada.

Each building represents a region and each region is provided with a gateway node. In our example, the gateway node is a mica2 mote connected to a PC. The routing agents are located in cooperative intelligent distributed systems (CIDS) lab on the base station, CAR 205. The sensor nodes are placed in rooms CAR 406 and CAR 410, as shown in Fig. 4.

The sensor nodes for HSH region are placed in HSH 402 and HSH 403. For simplicity, we placed four mica2 motes in each of the rooms. Three different conditions are considered in each of the region corresponding to CAR and HSH and described as follows: normal room temperature, higher than normal room temperature, and lower than room temperature.

The interface agent is running on a PC with IP address 192.162.166.192 located in room CAR 205. The receiving process of interface agent is managed by a web server, which is implemented in Java. For each incoming user request, an instance of the request-handler process is created in a separate thread.

The query plan along with the query arrives at the routing layer through gateway communication node. The routing layer's cluster agents disseminate the query packets as requests to the nodes on data collection layer to extract the data for the network.

The data collection layer gathers the data and passes it to the cluster head agents based on the highest residual power. These agents analyse the network data to maintain consistency and filter inaccurate and unwanted data. The data packets are then sent to the routing layer. The routing agents on the routing layer perform multi-target data aggregation; the aggregated results are sent to the interface layer where the interface agent makes the data available for the user and displays it on a web-based interface.

One of the mica2 motes, which acts as a base station, is connected to the PC located at CAR 205; the current IP of the base station is 192.162.135.5. The regional agents for CAR and HSH are located on the base station.

6. Implementations

We implemented a web-based prototype application to monitor the environmental conditions of a campus building described earlier. This prototype application will be extended for sandstorm monitoring. In this section, we describe the implementation details that will assist in the design of future work. The client and server software as well as the Web applications are described as follows:

Client Mote Software

When the motes are first started they directly transmit to the base station with a poll frequency of one minute. This is a relatively neutral starting state and will be needed until the routing agent transmits the first broadcast that contains the query execution plan and the routing information. Once a query is received from the base station, the motes start using the query execution plan. The query packet contains information such as poll frequency, poll count and threshold values for the sensors. During the duration of the query, the radio is turned off when not transmitting to conserve power. This means that no other queries can be received until the current active query is completed. This achieved at the medium access control layer [20,21].

Server Software

The application comprises three main components for the server software. First, the DBMS stores the data received from the client motes. Second, Apache Tomcat servlet container hosts the web application. Finally, TinyOS 2.0 [23]. environment of the connected mote acts as a base station to receive data from the client motes.

Web Application

The web application indicates the potential requirements for a WSN-based application. It is by no means authoritative; however, it provides a simple query interface and graphs, as well as tables to display the data received by the client motes.

7. Conclusion

This paper described the architecture of an intelligent software agent and discussed the cross layer optimization issues for query processing. This paper also provided the software design and components for a Web-based continuous monitoring application. A web-based wireless sensor network was implemented to monitor temperature, light, and mobility in two buildings in Acadia University. Our future plan is to develop a prototype system with the ability to trigger alerts to drivers and concerned authorities about the road conditions in case of sandstorms or heavy dust on highways.

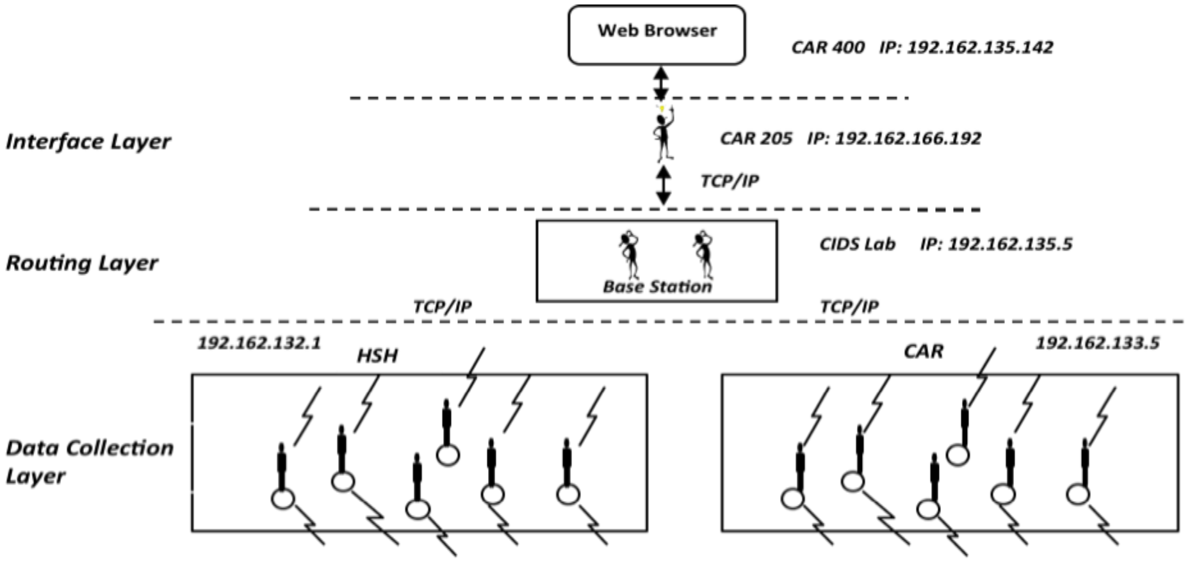

Fig. 4. Locations of all agents and sensor motes

Acknowledgments

The authors would like to thank NSERC and Acadia University for their support. The authors would also like to thank King Fahd University of Petroleum and Minerals (KFUPM) for their support through KACST أت 71 - 29.

References

[1] D. Estrin, D. Culler, K. Pister, and G. Sukhatme, Connecting the physical world with pervasive networks. *IEEE Pervasive Computing*, 59-69, January-March 2002.

[2] D. Estrin, R. Govindan, J. Heidemann, and S. Kumar, Next century challenges: Scalable coordination in sensor networks. In Proceedings of the International Conference on Mobile Computing and Networks (Mobi-Com), 1999.

[3] R. Tsolmon, L. Ochirkhuyag and T. Sternberg, Monitoring the source of trans-national dust storms in north East Asia. International Journal of Digital Earth, vol. 1, no.1, Taylor & Francis, pp. 119-129, 2008.

[4] A. Goudie, N. Middleton, Desert dust in the global system. Earth Science, Berlin, Springer, 2006.

[5] E. Shakshuki, H. Malik, M. Denko, Software agent-based directed diffusion in wireless sensor network, Journal of Telecommunication Systems, Springer, vol. 38, pp. 161-174, 2008.

[6] D. Culler, D. Estrin, and M. Srivastava, “Overview of sensor networks,” IEEE Computer, vol. 37, no. 8, 2004.

[7] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, “Wireless sensor networks: a survey,” Computer Networks, vol. 38, no. 4, pp. 393-422, March 2002.

[8] J. Lanier., “Agents of Alienation”, Hotwired debate with Pattie Maes, Vol. 2, pp. 66-72, New York, NY, USA, 1995.

[9] S. Russell, P. Norvig, Artificial Intelligence: A Modern Approach, Prentice Hall, 2003.

[10] S. Chandrasekaran, C. Dinesh, A. Murugappan, “An Artificial immune networking using intelligent agents”, Proceedings of the World Congress on Engineering, vol. 1, WCE 2008, July 2 - 4, 2008, London, U.K., 2008.

[11] J. Hill and D. Culler, “Mica: a wireless platform for deeply embedded networks,” IEEE Micro, vol. 22, no. 6, pp. 12-24, 2002.

[12] S. Madden, M. J. Franklin, J. M. Hellerstein, and W. Hong. Tag: A Tiny AGgregation service for ad-hoc sensor networks. Proceedings of the Fifth Symposium on Operating Systems Design and implementation (OSDI ’02), Boston, MA, USA, vol. 36(SI), pp.131–146, December 9 - 11, 2002.

[13] S. Harshavardha and C. Kishnendu, “Location-aided flooding: an energy-efficient data dissemination protocol for wireless sensor networks,” IEEE Transactions on Computers, vol. 54, no. 1, pp. 36-46, 2005.

[14] A. Lim “Distributed services for information dissemination in self-organizing sensor networks” Journal of the Franklin Institute 338, pp. 707–727, 2001.

[15] W. Heinzelman, A. Chandrakasan, and H. Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. In *Proceedings of the Hawaii International Conference on System Sciences*, Jan 2000.

[16] S. Bandyopadhyay and E. J. Coyle. An energy efficient hierarchical clustering algorithm for wireless sensor networks. In *Proceedings of the IEEE Conference on Computer Communications (INFOCOM)*, 2003.

[17] S. Jin, M. Zhou, and A. S. Wu. Sensor network optimization using a genetic algorithm. In *Proceedings of the 7th World Multiconference on Systemics, Cybernetics and Informatics*, 2003.

[18] K. P. Ferentinos, T. A. Tsiligiridis, and K. G. Arvanitis. Energy optimization of wirless sensor networks for environmental measurements. In Proceedings of the International Conference on Computational Intelligence for Measurement Systems and Applications (CIMSA), July 2005.

[19] E. Shakshuki, H. Ghenniwa, and M. Kamel. Agent-based system architecture for dynamic and open environments. International Journal of Information Technology and Decision Making, vol. 2, issue 1, pp. 105–133, 2002.

[20] E. Shakshuki, H. Malik, T, Sheltami, "Multi-agent Based Clustering Approach to Wireless Sensor Networks", International Journal of Wireless and Mobile Computing (IJWMC), Inderscience Publisher, vol. 3, no. 3, pp. 165-176, 2009.

[21] H. Malik, E. Shakshuki, T. Sheltami, "Agent Based Analytical Model for Energy Consumption among Border Nodes in Wireless Sensor Networks", 11th International Conference on Network-Based Information Systems, Lecture Notes in Computer Science, Springer-Verlag, pp. 189-201, Turin, Italy, September 1-5, 2008.

[22] I. Stojmenovic, "Localized network layer protocols in sensor networks based on optimizing cost over progress ratio", IEEE Network, vol. 20, no. 1, pp. 21-27, 2006.

[23] TinyOS, http://www.tinyos.net/

[24] Apache Tomcat, tomcat.apache.org/