Autonomous Sensor Network Architecture Model

András Tóth, Ferenc Vajda

Autonomous Sensor Network Architecture Model

András Tóth[1,2], Ferenc Vajda[1],

Budapest University of Technology and Economics, Department of Control Engineering and Information Technology, Magyar Tudósok Körútja 2., H-1117 Budapest, Hungary
{totha, vajda}@iit.bme.hu
Budapest University of Technology and Economics, Healthcare Technologies Knowledge Centre, Egry József Str. 18, H-1111 Budapest, Hungary
toth@emt.bme.hu

Abstract. To implement a maintenance free and reliable sensor network is necessary in complex sensor systems nowadays. There are studies, which can control developers, who want to build an autonomous sensor network in a dynamically changing environment. The existing solutions have several drawbacks because of the diversity and complexity of the wireless type sensor networks. This paper introduces a sensor network architecture model, which is a good base to implement a simulation environment for the systems before implementing them. The introduced features make the possibility to fine-tune the operational parameters during the planning phase. In this way, the presented sensor network architecture is working based on the presented roles to provide its services.

Keywords: role based, wireless sensor networks, sensors, sensor architecture model, maintenance free sensor networks, dynamic sensor network environment

1 Introduction

Nowadays, one of the main problems in sensor networks is to solve the problems that arise from network and resources maintenance. We need to optimize the efficiency and reliability of the systems. The main goal of a sensor networkis to gather measurement data from the environment. From a processing system point of view, it is a loss to maintain this kind of network. If the sensor network can handle these functions alone, then the processing network operation can be more reliable. There are two different control types. One, is a centralized control method, where the processor entities provide network guidance. These networks are more common than a non-centralized system, because the maintenance is taken on one side only, so problems and network errors are easier to recognize and repair. Therefore, most solutions use centralized control method based configuration.

In the current literature there are several system models and maintenance algorithms presented, which can handle large sensor networks. The main goal of these models is to verify system channels and other items in a fast and reliable fashion. The center of the system is usually one element of the sensor system or resides in the processing system and controls the network items using a selected administrator. In wireless sensor networks the relationships between entities (nodes) are changing dynamically. It is very difficult to use a static administrator entity, because some items often become disconnected and hidden within the environment. Developers and researchers implement many algorithms [2][3][4] and functions to search and reconnect these “lost” items. There are numerous methods to recognize these events. These solutions often mean additional work for the sensor administration and system maintenance side. It is necessary to poll the system elements frequently to see if the items have disappeared. If we could migrate this functionality to the entity side itself, these problems can be solved without adding overhead to the main system maintenance process.

Similar to a human community, it is normally expected that problems occurring can be solved by local members. If they cannot solve the problems, then the problems should escalate to a higher level, which can be, in this case, the system level. In this way we reduce the tasks and area of responsibility of the main system.

General weaknesses of the wireless sensor networks are the energy and communication reliability. The communication channels are often not static, so the problems with timing and energy management are much more complex. The relationships and network parameters are dynamically changing. If an element within the sensor network has to solve extra tasks, then extra energy is needed to be provided. If these items have spent all of the resources allocated to them, then their tasks should not be unfinished or incomplete.

There are studies [1-8], which introduce novel methods and algorithms to solve the above mentioned problems. These methods are designed to provide an optimum between energy consumption and usability. There are some problems, which are not answered by these solutions. It is important to show that the problematic functions do not belong to the processing system or an administrator center, because one of the weakest point of the entire system could be that element itself. If the system contains the logic presented above in a distributed way, and most of the elements can recognize and handle these problems, then a more reliable solution is given. In addition, the reliability and speed of processing can be also increased.

This paper introduces such a novel sensor system architecture model that solves the problems associated with maintenance roles. A “role based” solution translates this subject in very different way. In this case, roles solve only a part of problems within the local sensor network. Only one operation is used for sensor data gathering, there are many others, which are dedicated to validate communication channels and routes and provide energy/resource management and maintenance sensor elements within separated parts of the environment.

This sensor network model focuses on wireless solutions, because that topic is more complicated and needs new/better solutions, however, wired networks can also benefit by using this model in implementation of operational functions.

The presented work is connected to a novel scientific program of the "Development of quality-oriented and harmonized R+D+I strategy and functional model at BME" project. This project is supported by the New Hungary Development Plan (Project ID: TÁMOP-4.2.1/B-09/1/KMR-2010-0002).

2 Basic System Features

Our model can give a good base to build a simulation environment for the development process of a new sensor network. The work presented is not a simulation environment implementation. This model gives role based system architecture skeleton in a theoretical way. This system would provide the criterions mentioned before. Next sections will introduce the parameters and idea of this architecture model.

As mentioned before, the main goal of a sensor based system is to collect sensor information. A long-term usage network has many energy limitations. This forces engineers to develop more and more new algorithms, which make it possible to operate a network with higher reliability for a longer time with less energy.

In wireless sensor networks the parameters, which describe the environment of the sensor and the sensor itself have randomness and dynamism. This unpredictability means that the connections between the senders and receivers and their environment are not static. The neighborhood of the elements and the radio communication field can change all the time. Therefore, network searching and “keep alive” messages are necessary for reliable operations. However, the energy usage is high with every signal sent and received. It is also difficult to ensure that messages arrive to the target. Moreover, very important data security questions are arising.

First take a little look at relationships. Any connection between two network elements is temporary. For example, the physical movement element (node) can cause this phenomenon.

The provision of the available bandwidth in the network is very important. Of course, it is also important to prove that the communication routes are as short as possible and as “cheap” as possible. The system can reduce the needed bandwidth if the processing logic is partly placed within the sensor network itself. The only way this is possible, if the sensor network elements (nodes) has sufficient processing capacity to provide some preprocessing steps with raw sensor data.

Nowadays, sensor systems need to communicate with other networks. It can provide much higher efficiencies, if all members of these connected systems can use shared functions. This solution is possible if the system algorithms are well known and based on the same functional model. There are several research works, which describe a reliable base for future network standards [1-8].

In a wireless sensor network environment, it is important to consider data and information security issues. The system administrator functions needs to verify that sent messages are arrived to the original requesters and the receivers are always trusted.

In the network architecture model of the current work, it is necessary to design all of these requirements at the level of the proposed model.

3 Design Trends

The earlier models [9] were based on some specific features of the sensors and the measured environment. Hardware limitations will determine the plan and the operation of the system. Almost all of these early systems are centralized, because it is easier to implement the defined functions and processes.

Nowadays, algorithms are controlled and developed by system informatics and mathematics. More computing capability is needed, as well as, logical resources within the sensor network nodes. This is possible because, currently hardware components are less expensive and more reliable due to better engineering, technology innovations, mass production and economies of scale. The newest sensor systems are focusing on energy consumption and long-term reliable operation. To reach these goals the system has to reduce the radio communication frequency and the actual size of the messages. The size of messages can be reduced with preprocessing and filtering.

The newer sensor systems are closer to the PC based networks. The new addressing, messaging and processing methods illustrate a good example of how to plan a new sensor network, function and architecture.

Database Style Service Based Networks.

In these solutions the wireless and wired sensor networks operate like databases. There is a special query language defined to get information from the sensor system. There is also a preprocessing layer, which translates the queries to control or maintenance messages in the network. This system is a demand driven architecture. In this case all of the functions are centralized. The system is focusing on ease of handling and implementation. The main goal is to hide the sensor system specialty and special functionality from the higher service layers. This could be very useful, if we do not want to see what is happening in the network, while we are working with it.

Task Oriented Solutions.

As in the database type systems, the task allocation process is dynamic in the network. The main feature of the system is a location based task allocation. The system can describe which part of the system will provide sensor information. This solution has some disadvantages. The system requires homogeneous sensor network items. So, it can exchange data with any other entity in the network. In distributed networks it is very significant limitation and it means the main issue. [6]

Service Oriented Networks.

The service oriented solutions has an inner and an outer control. This is the one of the main advantages, because these systems can operate independently from the administrator functions of the central control system. [1][2] But it is also possible to instruct the sensor networks entities from outside of the closed sensor network. There are functions for monitoring and verifying the system processes. This architecture successful in distributed sensor environment. The members of the network can handle and respond to the local problems independently, without the control of a central administrator.

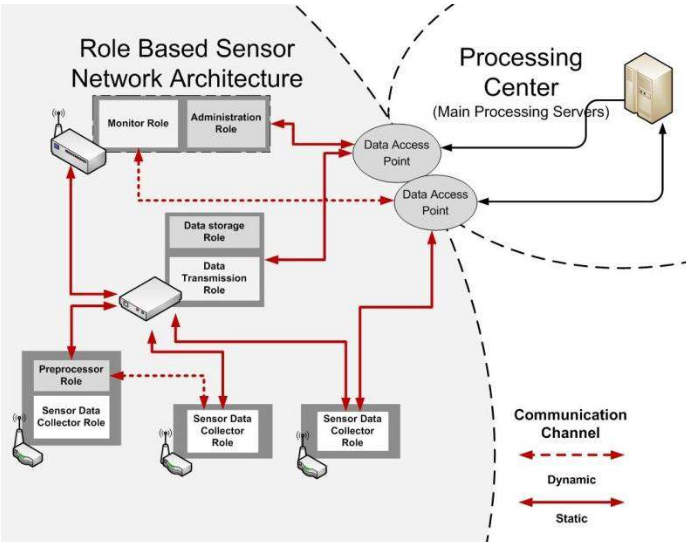

Figure 1: Role Based Sensor Architecture Model

Role Based Sensor Networks.

This approach is close to a service based solution. The main system has a built in operation process. This process has tasks, which are important for gathering information and maintaining the network. The tasks, which are related to one of the services in the system, are organized in a group. This group belongs to one role, which can be placed in any of the devices which is symbolized with an entity in the model. So the system maintenance is separated from the processing part (**Figure 1**). All of the roles have their own scope. The administrator role is able to send the control messages to the network, inside its scope or outside of it. It can also monitor and repair the system, but it can succeed only through maintenance messages for the special roles in the network. It is not necessary to know, where the processing roles are located. The system processes these queries, as it does with messages from the other roles.

One entity can provide more than one role, but it is limited by its resources. From the upper levels of the application, the sensor network seems to be one big sensing entity. This one entity is able to receive commands from the processing network, and able to serve monitoring information for the system control processes. From this information a system administrator can identify problems in the network.

The other main advantage is dynamism and modularity. The application code of the role can be run on all of the entities within the network. More roles function can run on the same entity parallel if it has enough resource. This roles location can be changed in the network as the optimal allocation needs. The administration role can control the displacements of roles in the network. One of the main criteria is to reduce energy consumption. Therefore, a solution that uses this model can be more energy efficient.

If one of the entities is running out of resources, then the nearest administration role should recognize this and try to select another place to run the needed role. This election is driven by the status of the known sensor parameters of entity. The developer can describe which roles are necessary in the sensor networks to work efficiently at the planning phase. The algorithm mentioned before, makes the system more reliable, because it will keep “alive” the roles of the system.

Two systems can cooperate with each other, because they have the same operation and protocol model. The difference could come from implementations, which can be solved by a translation protocol.

This version of the sensor system solutions can give more advantage than the others. It is more dynamic in operation. The functions solvers can change all the time and the system remains useful. The system can achieve its reliability in more cases than the others. The other advantage is to provide an energy efficient operation.

4 Roles Of The System

There are some base roles in the system. This set can be completed with implementation and simulation specific roles. These can make harder to plan cooperation between them. The root roles are dedicated to proper operation of the architecture. This solves the main tasks in the network. The models of the roles will be a set of usable applications and parameters in a specific configuration which can be run by the system members. The network members are different from each other and there are different versions from roles for the various types of entities. If the functions of the role could be modeled in one of the algorithm description language, then the running application modules can be generated after the planning phase.

All of the participants have a rating, which shows the capabilities for the running roles. This is the base of the implementation and the simulation side of the system. It describes the computing capacity and free resources. These can exist in a temporary storage, sensor item, extra logical unit or any other entity.

As stated, a critical criteria is the provision of an energy efficient operation. Communications consumes the majority of the overall energy. In this solution the units communicate with the nearest proximity entities. So the signal strength can be less than as it is at other network types. This network is using the “multi-hop” method at the higher layers, because members interact with their actual neighborhood. But in a bigger cluster the administration role gather data through its neighbors too. Alternatively, the administration processes can change a role location if the owner device is getting out of battery. Other devices can only nominate themselves for running the role if they have enough energy.

If the connection was lost between the two components, the administrator will start a new role owner selection. If the old owner device recognizes the lost connection, it will give up all of the services which belong to the actual authority.

If there is no usable role owner in the cluster, but it is necessary for the network, then the administrator module can transfer its task and request a new administrator runner election. If it is possible they will establish another cluster with other entities. If this does not give a reliable solution, then the role is going to get back to the processing system. After this, the process can be simulated until there is no appropriate entity to run it.

4.1 Basic Roles

The basic roles of the model are necessary to solve the maintenance and operational problems in a specific simulation environment. These contain the main rules for controlling and monitoring their neighborhood. The specific roles which come from the actual dependencies should implement near these. These entities provide the backbone of our sensor network.

- **Sensor data collector role:** This role symbolizes the functions of sensing with measuring. An entity with this role can measure one of the parameters in the environment. These results can be converted after collecting all of them. Any entity can take on this function if it has a sensor and a basic preprocessing unit. These criteria are very basic requirements in sensor networks nowadays. (**Figure 3.b**)
- **Data transmission role:** This role works like routers in a PC network. It has to transmit given data packets which arrive from a sensor data collector role or from another data transmission role. The destination state depends on the parameters of the data packet. E.g.: If the packet needs to be preprocessed somewhere, then the destination will be a preprocessor role, or a data transmission role, which is next to the closest preprocessor role.
- **Preprocessor role**: This role has the function for preprocessing data directly. In addition to this, it can also provide a basic filtering process to recognize that there are unnecessary information in the raw sensor data. If we use filtering we can reduce the number of transmitted data packets. With preprocessing we can reduce the packet sizes in any given period of measurement. This processing step means specific energy cost. So, the actual implementation will decide the level of the processing logic.
- **Data saving role**: In a wireless sensor network, it is a usual problem to lose connections between the sender and receiver. This connection should rebuild after a short period of time. Much depends on the features and the density of the network. If the communication connector does not have a usable route in direction to the destination role, then it can look for a storage role, which can temporarily store the data packets until the connection is made. In close proximity to this role there should be a data transmission role, because the stored message needs to be transmitted later.
- **Data access point**: This role symbolize that device which is directly connected to the processing server side of the system. If there is more than one access point in the network, then the effective bandwidth can increase.

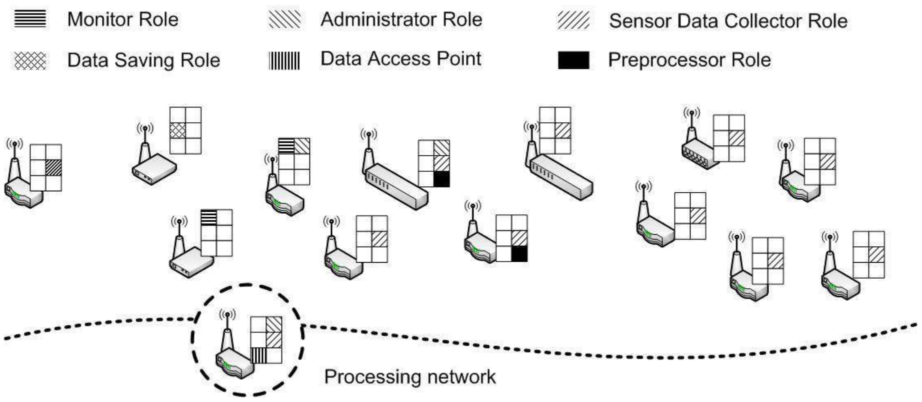

Figure 2: Distribution of the roles in a test design of the network

- **Monitoring role**: This role is making snapshots about their neighborhood and transmits them to the process server of the system. The monitor data packets are similar to the sensor data ones in the network. The processing network will recognize and translate this information. These statistical trends and information can give a good base for upper level system maintenance in an implementation. The developers and administrators can follow the network operations and errors, which can be repaired.
- **Administrator role**: This role is a key feature of the system. It can monitor all other roles and available entities in its authority. Its optimal location in the network depends on the other roles in the network. Of course since it is the most complex role in the network it needs the most complex resources. This role has to be online all of the time, because it has to provide the maintenance processes. If there is no entity left to handle this role, then it is goes back to the process center.

These roles provide the system features. These features are mobility and reliability. The model can give a more dynamic and adaptive solution for sensor systems. (e.g. in **Figure 2**) In large sensor systems there are a lot of usual problems with moving devices and the changing connections. At the start of the planning phase the developers have to imagine the system features before something is placed in the network. This system offers a good solution to simulate the members of the system before implementation. After evaluate a simulation from the theoretical model, then the developers can modify and optimize their composition. They can also plan the system for a longer term operation. The entity has to provide some own monitoring functions. It has to start its roles and functions regard to its features and parameters. (**Figure 3.a**) In order to ensure reliable operation for providing roles it has to always monitor the available resources. If it is necessary the entity can change its configuration independently (**Figure 3.c**).

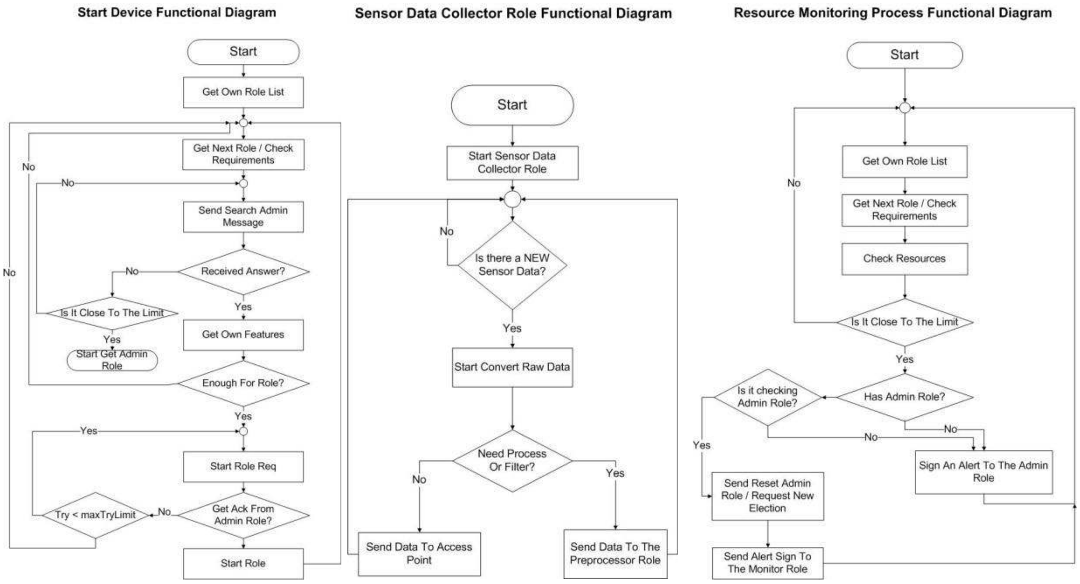

Figure 3: a.) Role starter flowchart, **b.)**Sensing function flowchart, **c.)** Resource maintenance function flowchart

5 Dynamic Features

Within a wireless sensor system, there are many dynamic parameters. The sensor devices in the network can move and disappear at any time during operation. The mobile devices also have various changing parameters. If they are running roles then its power will become low with time elapsing. If they have no energy for this role, then they have to give the role up and forward it to the administrator process. The entities can thus handle any energy problems in a more reliable manner. Using these algorithms, all of the roles and processes are safe. The central logic can assign the functionality to the sensor network system and retrieve it when necessary. The developers can decide which level they would like to implement system logic within the sensor system which is using the role based model for developing.

Energy Features: The system can dynamically change system role configurations. Every entity has a list of possibilities and capabilities. These are based on the implemented functions of the symbolized device. But there is a very important parameter, which is the power buffer size. If the power level gets too low for a role, then the role has to be moved to another entity. The monitor role collects information about the systems energy level. The administrators can receive a signal if the power level is too low. The maintenance staff should then correct and repair the problem. After a real system startup, devices can determine an optimal energy spending algorithm. In optimal cases the system power level matches the rate of all devices in the network.

Scalability: If the developers plan a large sensor network in a large environment, they should use more sensors in smaller sensor networks. These will provide many roles to perform system operations. In many cases the solutions could be solved with fewer sensors. This architecture model would be a good solution since the required number of roles could be reduced. If an entity has exhausted its resources then it would return any running roles. These roles keep functioning within the central core of the system. If the system needs any roles then numbers can increase with new entities as they join. In this case, the administrator sees this new requirement as a failed transmission or network errors. Then the administrator can allocate more roles dynamically to the network. For example, if two networks are connecting in one environment, new router roles will appear on the boundary of thems. Of course the sensor data collecting roles cannot go to the processing center, because it needs a feature what must stay with the sensor entity. If they cannot do their sensing work they have to enter a suspend state upon receiving an alert message.

Autonomous Operation: In other service based models it is necessary to be a member, which can see the entire image of the network. This element stores something like service library. This list shows the entities, which belong to the actual services.

In this works case, the system could not see this image. The administration module sees its neighborhood. The members of a cluster or a group can reach an administrator role. They can address their messages with features. If they need a task from a specific role, then they send it to a route which contains the needed one. The role can disappear from the route, because it has been moved somewhere else. But on the route, the original owner exists, which had the old role and can give a new forward direction for message.

The system has to change a role location only if necessary. This change could be required for reasons of availability and/or energy needs. These movements are taken autonomously by the roles and entities. If one of them senses that a requirement for running a role is ending then it will release it and assign it to the closest administration logic. The roles can handle these problems within their local authority.

If an entity senses that, it has been removed from an environment, and it cannot reconnect to peers, then it will look for new connections with visible ones. It will request a link to the closest administrator role. The visible entities access the service list, discover what they can serve and route to them. This algorithm solves autonomously connection problems, and can enlist new elements to the network.

6 Advantages Of The Special Architecture Model

This model is a good base for the simulations and the modifications a sensor network before develop any hardware or software component. There are a few parameters, which have to be described for the model in the future, but with the exception these, the developers can plan and simulate the system easily after complete the model and calculate the parameters. These parameters are useful at a validation process too. With this platform the maintenance staff can follow system operations and events. The monitor role provides its data source.

During the planning of a project this model provides online simulation parameters to modify and optimize the system from the beginning. The developers can calculate the lifetime, processing and storage resources. The sensor network can give a reliable answer for connection errors and the operation deviances.

7 Future Work

There is a need to develop a newer version of the evaluation algorithm which is able to describe the roles location more efficiently. At this stage the location can be changed, if the requirements of connection change. We would like to integrate more points of view into the processes. The architecture can be more energy efficient and reliable if there is an independent process in the network which can collect information for this purpose. We will implement the specific features into an open source version, which will be usable all PC platforms. After this, we would like to test and upload it to a public site. It can be useful for engineers around the world and they can contribute their knowledge. This information is necessary to continue developing this architecture model. We will integrate the model simulation phase to existing sensor networks development process. We would like to measure the differences between the “typical” planned and the “role based” planned solutions.

Acknowledgement.

I would like to thank my consultant Ferenc Vajda Dr. for monitoring and checking my work during the development phase of the model construction. Also many thanks to my supervisor Lóránt Vajda for the technical environment and the support meetings concerning this topic. And I am very grateful for the technical questions and the final proofreading of Randall Claywell, who also checked my research’s direction.

This work was supported by TÁMOP-4.2.1/B-09/1/KMR-2010-0002.

8 Literature

1. Lombriser, C., Marin-Perianu, R., Roggen, D., Havinga, P., Tröster G.: Modeling Service-Oriented Context Processing in Dynamic Body Area Networks, Vol. 27, NO. 1, IEEE Journal, (2009)
2. Casola, V., Gaglione, A., Mazzeo, A.: SeNsIM-Web: a Service Based Architecture for Sensor Networks Integration, IEEE, (2009)
3. Jiong Jin, Wei-Hua Wang and Palaniswami, M.: Application-Oriented Flow Control for Wireless Sensor Networks, Third International Conference on Networking and Services(2007)
4. S. Han, R. Rengaswamy, R. S. Shea, E. Kohler, and M. B. Srivastava: A Dynamic Operating System for Sensor Nodes, In Third International Conference on Mobile Systems, Applications and Services (2005)

5. Bergstrom, E., Pandey R.: Composing µSIC: A Lightweight Service Model for Wireless Sensor Networks, International Conference on Sensor Technologies and Applications(2007)
6. Kushwaha, M., Amundson, I., Koutsoukos, X., Neema, S., Sztipanovits, J.: OASiS: A Programming Framework for Service-Oriented Sensor Networks, IEEE(2007)
7. Yuebin Bai, Haixing Ji, Qingmian Han, Jun Huang, Depei Qian: MidCASE: A Service Oriented Middleware Enabling Context Awareness for Smart Environment, International Conference on Multimedia and Ubiquitous Engeneering(2007)
8. Victor Foo Siang Fook, Jayachandran Maniyeri, Aung Aung Phyo Wai, Pham Viet Thang and Jit Biswas: Service Oriented Architecture for Patient Monitoring Application, IEEE International Conference on Industrial Informatics(2006)
9. Muruganathan,S.D. Ma, D.C.F. ; Bhasin, R.I. ; Fapojuwo, A.O.: A centralized energy-efficient routing protocol for wireless sensor networks, Communications Magazine, IEEE (2005)