MANNA: A Management Architecture for Wireless Sensor Networks

Linnyer Beatrys Ruiz, Federal University of Minas Gerais and Pontifical Catholic University of Paraná
José Marcos Nogueira and Antonio A. F. Loureiro, Federal University of Minas Gerais

ABSTRACT

Wireless sensor networks (WSNs) are becoming an increasingly important technology that will be used in a variety of applications such as environmental monitoring, infrastructure management, public safety, medical, home and office security, transportation, and military. WSNs will also play a key role in pervasive computing where computing devices and people are connected to the Internet. Until now, WSNs and their applications have been developed without considering a management solution. This is a critical problem since networks comprising tens of thousands of nodes are expected to be used in some of the applications above. This article proposes the MANNA management architecture for WSNs. In particular, it presents the functional, information, and physical management architectures that take into account specific characteristics of this type of network. Some of them are restrict physical resources such as energy and computing power, frequent reconfiguration and adaptation, and faults caused by nodes unavailable. The MANNA architecture considers three management dimensions: functional areas, management levels, and WSN functionalities. These dimensions are specified to the management of a WSN and are the basis for a list of management functions. The article also proposes WSN models to guide the management activities and the use of correlation in the WSN management. This is a first step into a largely unexplored research area.

INTRODUCTION

Wireless sensor networks (WSNs) provide distributed network access to sensors, actuators, and processors embedded in a variety of equipment, facilities, and the environment. A WSN represents a new monitoring and control capability for applications such as environmental monitoring, infrastructure management, public safety, medical, home and office security, transportation, and military [1–4]. A WSN combines micro electromechanical systems (MEMS) technology, new sensor materials, low-power signal processing, computation, and low-cost wireless networking in a compact system. Currently, it is possible to find sensor nodes varying from a few millimeters to 2 m. Advances during the last decade in integrated circuit technology have enabled the manufacturing of far more powerful but inexpensive sensors, radios, and processors, allowing mass production of sophisticated systems connecting the physical world to computer networks.

The large use of WSNs depends on the design and development of a scalable, low-cost sensor network architecture. Such applications need to send sensor information to users or network entities at a low bit rate using low-power transceivers. Continuous sensor signal processing enables the constant monitoring of events in an environment in which possibly a few data bytes would suffice. Some of the applications foreseen for WSNs will require a large number of devices on the order of tens of thousands of nodes. Traditional methods of sensor networking represent an impractical demand on cable installation and network bandwidth. Performing the processing at the source can drastically reduce the computational burden on application, network, and management. On the other hand, any solution must take into account-specific characteristics of this type of network.

Until now, WSNs and their applications have been developed without considering a management solution. This may not be a problem for small networks, but will definitely be when applications, in order to work properly, will need to reconfigure and adapt themselves based on information scattered over the network. This article proposes a management architecture for WSNs. In particular, it presents an information architecture and a functional management architecture that take into account-specific characteristics of this type of network.

Management of WSNs is a new research area that only recently started to receive attention

This work is partially supported by National Research Council CNPq, Brazil.

from the research community. In this sense, this work presents a contribution to the field, since it proposes a WSN management architecture. We present a separation between both sets of functionalities (i.e., application and management) through a management architecture for WSN. This will make possible the integration of organizational, administrative, and maintenance activities for this kind of network.

The rest of this article is organized as follows. We present the main characteristics and metrics of WSNs. We then discuss the important aspects in the management of WSNs. We present and discuss the MANNA management architecture for WSNs, as well as a possible management situation and how the MANNA architecture works. Finally, we present our conclusions.

WIRELESS SENSOR NETWORKS

Sensor nodes in WSNs are spread over a region and communicate among themselves using point-to-point wireless communication, possibly forming an ad hoc network. Sensors collect, process, and send data observed from the environment to other nodes. Basically there are three types of nodes: *common nodes* responsible for collecting sensing data, *sink nodes* responsible for receiving, storing, and processing data from common nodes, and *gateway nodes* that connect sink nodes to external entities called *observers*. WSNs can also include actuators that enable control or actuation on a monitored area.

The observer is a network entity or final user that wants to have information about data collected by sensor nodes. Depending on the type of application, the observer may send a query to the WSN and receive a response from it. A sensor element generates data about a given phenomenon. A WSN may collect different sensor data such as temperature, pressure, electromagnetic field, and chemical agents since it can comprise different sensor elements. A wireless sensor node comprises one or more sensor elements, battery, memory, processor, and transceiver. Programs developed to execute in a wireless sensor node must take into account its hardware restrictions.

A WSN is said to be *homogeneous* when all nodes have the same hardware; otherwise, it is *heterogeneous*. The nodes are *autonomous* when they are able to execute self-configuration tasks without human intervention. A WSN is *hierarchical* when nodes are grouped for the purpose of communication and *flat* otherwise. In a hierarchical network, it is common to have a base station that works as a bridge to external entities. A WSN is *static* when nodes are stationary and *dynamic* otherwise. Note that the topology may be dynamic even when nodes are stationary since new ones can be added to the network or existing nodes become unavailable. A WSN is *symmetric* when each transceiver has the same transmission range and *asymmetric* otherwise. A WSN is *continuous* when sensor nodes collect data and send them to an observer continuously along time and *on demand* when they answer to observer's queries. A WSN is *reactive* when sensor nodes send data referring to events occurring in the environment and *programmed* when nodes collect data according to conditions defined by the application. A WSN is *hybrid* when it has at least two of the above characteristics concerning dissemination of information.

WSNs have other important characteristics depending on the application. Some of them are coverage, accuracy, fidelity, density, self-organization, adaptation, and location. However, the points described above will play an important role in the definition of the functional architecture presented in this article.

When designing and evaluating WSNs for different applications, some of the metrics that should be considered, depending on the environment, are described below.

Longevity/energy: Energy is a critical resource in a WSN. Thus, all operations performed in the network should be energy-efficient. Network availability can be measured as the amount of time some or all sensor nodes in the network continue to obtain sensing data and pass them to the application.

Latency: This refers to the time interval between the instant the sensor gets the data and the moment they are delivered to the destination, and it has two components: inside the network, from sensor to sink node, and from sink node to observer. Depending on the kind of application and network latency, the data received by the observer may be of no value and should be discarded.

Accuracy: This indicates the reliability or exactness of a result. It can also be defined as the fraction of valid results from all results obtained. Factors such as environmental conditions when the data are obtained and communication range of the sensor node may also degrade accuracy. The application plays an important role in this metric since it is responsible for establishing the amount of energy to be spent in obtaining data. As a consequence, the network should adapt to the accuracy metric defined by the application and according to an upper limit of latency.

Fault tolerance: In a WSN, nodes may fail due to energy, physical destruction, communication problems, or inactivity (a node becomes *suspended*). Even if these situations occur, it may be desirable for the network to continue to operate properly.

Goodput: This is the ratio of the total number of packets received by the observer to the total number of packets sent by all the sensors over a period of time.

MANAGEMENT OF WIRELESS SENSOR NETWORKS

Traditional computer networks are designed to accommodate a diversity of applications. Network elements are installed, configured, and connected in a network in a way to provide different kinds of services. In general, management aspects are clearly separated from network common activities (i.e., the services they provide to their users). Therefore, it is said that there exists an overlapping of management and network functionalities, but the implementation can be thought of independently. In the following we

A wireless sensor node comprises one or more sensor elements, battery, memory, processor, and transceiver. Programs developed to execute in a wireless sensor node must take into account its hardware restrictions.

The initial configuration of a WSN can be quite different from what was supposed to be in the case of throwing the nodes in the ocean, forest, and other remote regions. In unpredictable situations, a configuration error may cause the loss of the entire network even before it starts to operate.

discuss important characteristics of WSNs that make their management different from a traditional computer network.

MANAGEMENT REQUIREMENTS

In computer networks, replacement of faulty components or resources by technicians is a normal fact. The network tends to follow well established planning of resources available, and the location of each network element is well known. In a WSN this is often not the case, since the network is planned to have unattended operation and nodes can be discarded, lost, and out of operation temporarily or permanently. In this scenario, faults are a common fact, what it is not expected in a traditional network. In fact, the initial configuration of a WSN can be quite different from what was supposed to be in the case of throwing the nodes in the ocean, forest, and other remote regions. In unpredictable situations, a configuration error (e.g., planning error) may cause the loss of the entire network even before it starts to operate.

Depending on the WSN application, it may be interesting to uniquely identify each node in the network. Furthermore, we may be interested in a value associated with a given region and not a particular node. For instance, we may be interested in the temperature at the top of a mountain. A WSN is typically data-centric, which is not common in traditional networks.

The objective of a WSN is to monitor and, eventually, control a remote environment. The objective of WSN management is to define a set of functions that intend to promote productivity, as well as to integrate in an organized way functions of configuration, operation, administration, and maintenance of all elements and services of a sensor network. Nodes execute a common application in a cooperative way (i.e., there is clearly a common goal in the overall network), which may not be the case in a traditional network.

PRINCIPLES FOR DEFINING A MANAGEMENT ARCHITECTURE

We propose that the WSN management be simple, adherent to network idiosyncrasies, including its dynamic behavior, as well as efficient in its use of scarce resources. In this work we consider the following principles:

- Try to resolve in an extensive way specific problems derived from the dependencies WSNs have on applications and energy restrictions. For example, location mechanisms can be different among them depending on environment and network organization.
- Build a generic management function list from abstractions of different functional areas, management levels, and network functionalities.
- Establish an open and documented information model that allows reuse of objects, and syntax and semantic uniformity of management information.
- Provide a functional architecture that considers generic configuration of a wireless sensor network.
- Adapt protocols, algorithms, and mechanisms already developed for wired and wireless networks.

Hence, the approach used in this development deals with complex management situations by decomposing a problem into smaller subproblems, in successive refinement steps. We work with each functional area, each management level, and propose a new abstraction level of WSN functionalities described later. As a result, we present a list of management functions next, independent of technology and functional architecture adopted.

SERVICE MANAGEMENT COMPONENTS

The definition of management services consists of finding which activities or functions must be executed, when, and with which data. Management services are executed by a set of functions. They need to succeed to conclude a given service.

Management functions represent the lowest granularity of functional portions of a management service, as perceived by users. This means that the management architecture must exhibit a function list to deal with the integrated functioning of a WSN, applications, and users. Therefore, management functionalities will be independent of network target activities, even when this is not apparent in the implementation.

The MANNA architecture establishes that the WSN management does not end in its functions, though. It is necessary to go further. Policy management will be dependent on network states. A network state, or part of it, can be viewed from different perspectives and varies with the moment. The MANNA architecture defines WSN models that represent aspects of the network, and serves as a reference to the management functions. These models provide an abstract vision of the system through which it is possible to hide all nonrelevant aspects of a certain objective.

To model the computing aspect of the management service, a MANNA architecture provides policy-based management. In the specification of these policies there are conditions that should be satisfied so specific functions are executed and thus provide the desired management service.

The conditions for executing a function are obtained from the WSN models. For example, a maintenance service of the coverage area obtains the energy and sensing range conditions of the nodes in the network, making use of some WSN models such as energy map and topology map. To find out sensing areas that are not monitored, the service executes the coverage area supervision function. In this way it obtains the information that allows it to choose the most appropriate policy to tackle this problem.

The relationship among services, functions, and WSN models is illustrated in Fig. 1. The figure represents a scheme to construct the management, starting at the definition of both services and functions that use models to achieve their goals. A service can use one or more management functions. Different services can specify common functions that use models to retrieve a network state concerning a given aspect.

Management Functions

Management functions can be *automatic*, when executed by some software invoked as a result of information acquired from a model; *semi-automatic*, when executed by a human operator assisted by a software system that provides a network model or invoked by a management system; and *manual*, when executed outside of the management system.

Five possible states are defined for a function: *ready*, when the necessary conditions to execute a function are satisfied; *not-ready*, when the necessary conditions to execute a function are not met; *executing*, when the function is being executed; *done*, when the function has successfully executed; and *failed*, when a failure occurs during the execution of the function.

A partial list of the management functions, in no particular order, is given below:

- Environmental monitoring function
- Monitored area definition function
- Coverage area supervision function
- Node deployment definition function
- Node deployment function [5]
- Environmental requirements acquisition function
- Network operating parameters configuration function
- Topology map discovery function
- Network connectivity discovery function
- Aggregation discovery function
- Node density control function
- Priority of action definition function
- Management operation schedule function
- Cooperation discovery function
- Synchronization function
- Energy map generation function
- Network coverage area definition function
- User interface function
- Self-test function
- Node localization discovery function
- Node operating state control function
- Energy level discovery function

We define some functions, listed below, that allow one to obtain characteristics related to the efficiency and effectiveness of a WSN. Some of these quantitative functions are defined to obtain parameters presented in [6]:

- Network settling time function
- Network join time function
- Network depart time
- Network recovery time function
- Frequency of updates (overhead) function
- Memory requirement function
- Network scalability function
- Energy consumption function

WSN Models for Dynamic Representation

In a WSN the network conditions can vary dramatically in time. In this case, the utilization of models established by MANNA is of fundamental importance for management, although its updating cycle can be extremely dynamic and complex. Based on the information obtained with these models, services and functions are executed according to management policies.

There are two kinds of management information: *static* and *dynamic*. Static information describes the service configuration, and both the network and the network element. It is mapped to object classes. The MANNA architecture defines an information model for representing static information. Dynamic management information is described by WSN models and needs to be obtained frequently. The acquisition of this information has a cost in terms of energy consumption. Therefore, an important aspect is to determine the adequate moment, frequency, and fidelity for updating that information. Furthermore, the information collected may be not valid at the moment it is processed by the management entity due to delays, omissions, and uncertainty present in WSNs.

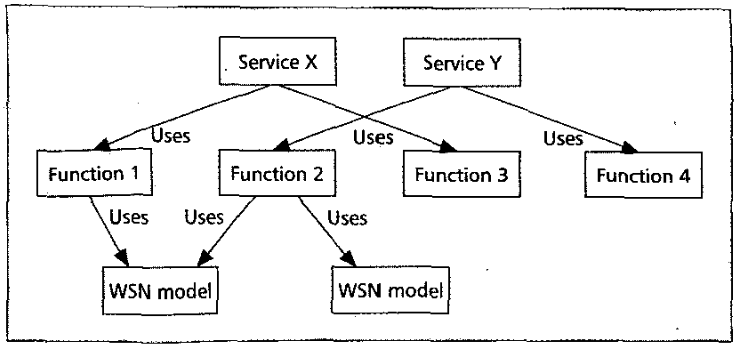


Figure 1. *Relationship among services, functions and WSN models in MANNA management.*

The dynamic information represented in the network models could or could not be stored in MIBs. Examples of dynamic models are given below:

Sensing coverage area map: Describes the actual sensing coverage map of the sensor elements.

Communication coverage area map: Describes the present communication coverage map from the range of transceivers.

Behavioral model: Represents the behavior of a WSN. Statistical and probabilistic models may be much more efficient in estimating network behavior than deterministic models.

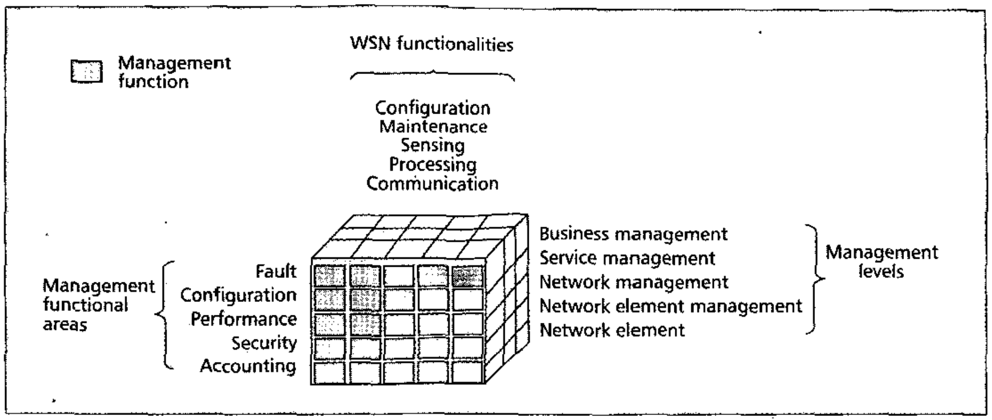
Dependence model: Represents the functional dependency that exists between the nodes. The network is modeled as a graph, where the nodes in the graph correspond to nodes in the WSN, and the edges between them represent the existing dependency relations (e.g., the connectivity between the nodes). In order to represent the dependencies, Bayesian or Markovian models, for instance, may be used.

Network topology: Represents the actual topology map and the reachability of the network. It may be used to obtain information about the necessity of adding new nodes [5].

Residual energy: Represents the remaining energy in a node or network. This information may also be available considering a region or time interval. Using this information, together with the data generated by the network topology model, it is possible to identify the areas that will have shorter lifetimes [5].

Usage standard: Represents the activity of the network. It can be delimited for a period of time, quantity of data transmitted for each sensor unit, or by the number of movements made by the target [5].

In WSNs, all operational, administrative and maintenance characteristics of the network elements, the network, the services, and business, as well as the adequate execution in the activities of configuration, maintenance, sensing, processing and communication are dependent on the configuration of the WSN.



■ **Figure 2.** *Management functionality abstractions.*

Cost: Represents the cost of equipment, energy, and personnel necessary to maintain the desired performance levels.

In telecommunication networks and distributed systems, there are two categories of relations: *structural* and *cooperational*, which may be represented through these models:

Structural models: Represent the relations of aggregation and connectivity between network elements, as well as the description of the same network elements.

Cooperational models: Represent relations of interaction between network entities. For example, there is a *service-user* relation. The relations of cooperation are created, activated, and terminated (*normally, abnormally, aborted*, etc.) between the network components and distributed systems. The components involved may, by their own initiative or activated by foreign actors, adjust their behavior or share resources, contributing to a common objective. In sensor networks, cooperation between the sensors, in general, is peer to peer. Only two sensor nodes cooperate with each other at a given moment.

WSN Functionalities as a New Dimension to Management

Traditional network management is organized over two planes, management functional areas and management levels. The MANNA architecture defines a new dimension to management. It is another abstraction level where the network functionalities are also considered. In this way, WSN management will have an organization that comes from abstractions offered by management functional areas, management levels, and network functionalities (configuration, maintenance, sensing, processing, and communication). The MANNA architecture considers the three abstraction planes in the definition of a management function.

Figure 2 presents the existing relationships in the definition and utilization of management functions. The new dimension introduced can be observed in the upper part of the figure.

Management Functional Areas

In the following, we present a contribution to WSN management technology from the perspective of functional areas.

The concepts involved with the functional areas of WSNs differ from established definitions for traditional networks or even other wireless networks. The MANNA architecture considers that the fault, security, performance and accounting functional areas are extremely dependent on the configuration functional area. In WSNs, all operational, administrative and maintenance characteristics of the network elements, the network, the services, and business, as well as the adequate execution in the activities of configuration, maintenance, sensing, processing, and communication are dependent on the configuration of the WSN. This idea is depicted in Fig. 3 where the configuration functional area plays a central role.

Configuration management is a functional area of high relevance in WSN management. Since the objective of a sensor network is to monitor (acquisition, processing, and delivery of data) and, eventually, to control an environment, any problem or situation not anticipated in the configuration phase can affect the offered service. Some management functions we have defined for network-level configuration management are requirements specification of the network operational environment; monitoring of environmental variations; size and shape definition of the region to be monitored; node deployment, random or deterministic; operational network parameters determination; network state discovery; topology discovery; network connectivity discovery; control of node density; synchronization; network energy map evaluation; coverage area determination;and integration with the observer. Some management functions we have defined for network-element-level configuration management are, node programming, node self-test, node location, node operational state, node administrative state, node usage state, and node energy level.

Faults in wireless sensor networks are not an exception and tend to occur frequently. This is one of the things that make management of WSNs different from traditional network management. Faults happen all the time due to energy shortages, connectivity interruptions, environmental variations, and so on. In general, sensor networks must be fault-tolerant and robust, and must survive despite occurrences of faults in individual nodes, the network, or even services provided. In addition to events caused by energy problems, other events can happen in a WSN related to communication, quality of service, data processing, physical equipment fault, environment, integrity violation, operational violation, security, and time domain violation. Therefore, even if a node has an adequate energy level to execute its function, it may decide not to do that because of other reasons.

Security functionalities for WSNs are difficult to provide because of their ad hoc organization, intermittent connectivity, wireless communication, and resource limitations. A WSN is subject to different safety threats: internal, external, accidental, and malicious. Information or resources can be destroyed, modified, stolen, removed, lost, or disclosed, and service can be interrupted. Even if the WSN is secure, the environment can turn it insecure or vulnerable.

Sensor networks have inherited the typical problems of wireless networks, including a high percentage of communication data loss and difficulty in controlling energy consumption. Two of the main objectives of *performance management* in a WSN are the quality of information acquisition and distribution services. In performance management, there is a trade-off to be considered: the highest the number of managed parameters, the highest the energy consumption and the lowest the network lifetime. On the other hand, if parameter values are not obtained, it may be not possible to manage the network appropriately.

Accounting management includes functions related to the use of resources and corresponding reports. It establishes metrics, quotes, and limits that can be used by functions of other functional areas. These functions can trace the behavior of the network, and even make inferences about the behavior of a given node. Some functions related to accounting management are discovery, counting, storing, and data reporting of a parameter; network inventory; determination of communication costs; energy consumption; and traffic checking.

Management Levels

In the logical layer architecture (LLA), management functionalities depend on the management level. Many traditional management systems use this model in a bottom-up approach. In the MANNA architecture, the LLA model is used in a top-down approach. After analyzing the business level issues, the necessities of the lower levels become clear. Similarly, it is only after defining the application, including the corresponding requirements on the service layer, that we can plan the network and network element management layers, and network element. This is a key observation when reasoning about the WSN management.

In the following we present a brief discussion concerning WSN management from the perspective of management levels.

Requirements that allow the characterization of a sensor network came from the objectives defined for the *business management layer*. Since WSNs depend on applications, business management deals with service development and determination of cost functions. It represents a sensor network as a cost function associated with network setup, maintenance, sensing, processing, and communication.

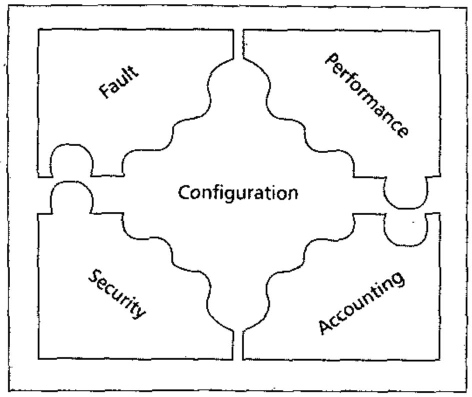


Figure 3. *The role of configuration management.*

The management of the *services* provided is the responsibility of the MANNA architecture. WSN services are concerned with functionalities associated with application objectives. A common priority for all services is to minimize energy consumption. Examples of WSN services are data gathering, processing, and communication.

In *network-level management*, relationships among sensor nodes are to be considered. It is known that individual nodes are designed to sense, process data, and communicate, contributing to a common objective. In this way, nodes can be involved in collaboration, connectivity, and aggregation relationship.

The *network element level* of the logical layer architecture corresponds to network elements that need to be managed or execute some management function. Considering that applications may require networks with a large amount of sensor nodes, network element management can deal with a group of nodes. In such a case, a manageable element can be a cluster of nodes or a cluster head rather than an individual node.

Management Architecture

The MANNA architecture comprises functional, information, and physical architectures. They are described below.

Functional Architecture

The functional architecture describes the distribution of management functionalities in the network among manager, agent, and management information base (MIB). In the architecture it is possible to have a diversity of managers and agent locations. The functional architecture suggests both locations for managers and agents and functions they can execute.

WSN Manager — The WSN management can be centralized, distributed, or hierarchical. In a centralized management network, there is a single manager that collects information from all agents and controls the entire network. A distributed management network has several managers, each responsible for a subnetwork and communicating with other managers. In a hierar-

In the logical layer architecture, management functionalities depend on the management level. Many traditional management systems use this model in a bottom-up approach. In the MANNA architecture, the LLA model is used in a top-down approach.

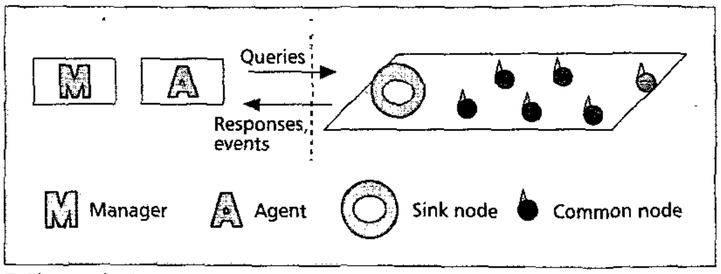


Figure 4. *Agent and manager located externally in the WSN.*

chical management network, there are intermediate managers to distribute the management tasks. The management alternative to be chosen depends on the application running on the WSN. In any solution, it may be important to have a manager entity located externally to the WSN.

WSN Agents Location — The development of a functional architecture raises some questions related to the location of agents. The most adequate location for an agent depends on the kind of WSN.

A first alternative for agent location is to place it close to the manager (i.e., external to the network). This would cause isolation of the management and make difficult to integrate it in the future and even access other management systems. This configuration can be viewed in Fig. 4.

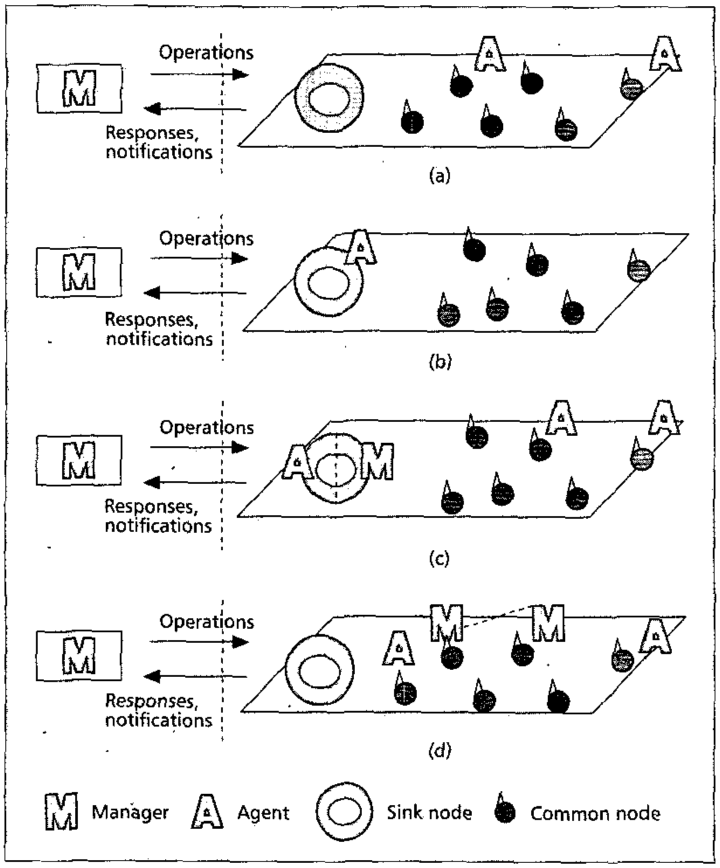


Figure 5. *Manager and agent location in flat WSNs.*

In the following we explore some possible configurations:

Agents in flat and homogeneous WSNs: A flat WSN has at least one sink node to provide network access. All network nodes have the same hardware configuration. Some possible alternatives for flat and homogeneous networks considering agent location in the WSN are:

- Agents in network and external manager (Fig. 5a).
- Agent in sink node (Fig. 5b).
- Agents and manager in network. The two possibilities for manager organization are hierarchical (Fig. 5c) and distributed (Fig. 5d).

In any of these proposals, the main concern is the large amount of traffic that may be generated in response to operation requests and sending notifications. Another alternative is to place managers inside the network, allowing them to communicate among themselves. This defines distributed management. If having agents as part of common nodes, some questions remain such as how to distribute the agents, how to define domains for the agents, and how to deal with nodes with more than one agent.

Agents in flat and heterogeneous WSNs: In a heterogeneous WSN, nodes differ in their hardware physical capabilities. Agents can be placed in more powerful nodes, as long as they present adequate location in the network. The sink node can host an intermediate manager or even present no management function at all. To establish distributed management, we can place agents in less powerful nodes and managers in more powerful ones.

Agents in hierarchical homogeneous or heterogeneous WSNs: In this kind of network, there is no sink node. A cluster head node is responsible for sending data to a base station. It also communicates with the observer. The cluster head may also execute correlation of management data. This computation may decrease the information flow and thus energy consumption. The correlation may also allow multiresolution where differences are filtered and higher precision is obtained. Some possible alternatives for a hierarchical WSN considering the agent location are:

- Agents in the network and external manager (Fig. 6a)
- Agent in the base station (Fig. 6b)
- Agents in the network and intermediate manager (Fig. 6c)
- Agents and distributed managers in the network (Fig. 6d)

Centralized management for WSNs, as well as for traditional ad hoc networks, is not always appropriate. One main reason is the traffic concentration problem, caused by a central manager that receives and originates management traffic. In addition, the *response implosion problem* may happen when there is a high volume of incoming replies triggered by management operations or events. In any case, there will always be one access point (sometimes more than one) through which data go to the observer or management application. The access point represents a sink node or base station that can make use of a gateway to communicate with the external environment.

To resolve the response implosion problem, one possibility is to select only a subset of agents to send replies back, known as *fidelity*. This approach may be suitable for densely populated sensor networks with a large number of sensor nodes, where missing information from some nodes can be ignored with acceptable accuracy. The accuracy of the calculation might significantly degrade in a sparse sensor network or one with a small number of nodes not collecting enough replies. However, the number of replies may not be small enough to be received without taking into account the response implosion problem. One solution is to make a scheduled response approach [7].

Management Information Base — The description of objects present in the information model and the relationship among them are specified in the management information base. In the WSN, to update an MIB with the current network state may require measuring various parameters. In general, the collection of these parameters may present spatial and temporal errors.

To have higher precision in the network state, probabilistic measures should be made with higher granularity. As in any probing, this would take a finite amount of system energy and could modify the network state. This is called the *probe effect*. In this way, better precision of management information requires modification of the state.

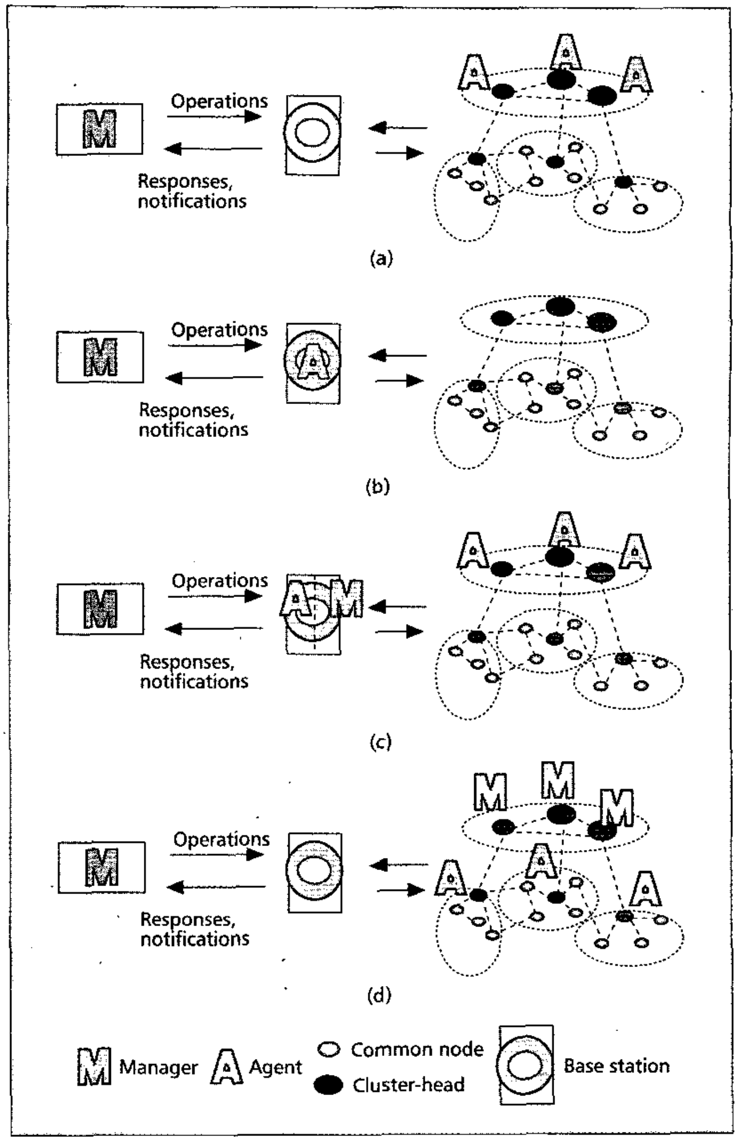
This work proposes limitation of scope as a method to reduce uncertainty and energy consumption while updating the MIB. *Spatial limitation* consists of defining a physical space where the data will be considered for management. *Temporal limitation* defines a time window (fixed or sliding) inside which the collected data are considered. *Functional limitation* selects the data of a certain functional network segment for management (e.g., the data of a group of nodes or a group leader).

PHYSICAL ARCHITECTURE

The physical architecture is the implementation of the functional architecture. In doing this, physical aspects such as the management protocol, the physical location of agents, agent functionalities, management service implemented, and supported interfaces for WSNs are defined.

The interface between the management entities should use a lightweight protocol stack. The MANNA architecture does not define a protocol stack for these interfaces, but provides protocol profiles that may be adequate for each application type.

Although the Simple Network Management Protocol (SNMP), Common Management Information Protocol (CMIP), WBM, and Ad Hoc Network Management Protocol (ANMP) [8] management protocols allow management in a decentralized form and event-oriented, the structure of the managed components is always too rigid. In these paradigms, the management intelligence always resides in the managing instance, while the information is generated in the managed instances. An alternative method would be the delegation of management functionalities to the managed systems. A solution, for supporting this feature in the implementation of the physical architecture is *management by delegation* (MbD). Other alternatives are to implement intelligent agents and mobile agents.



■ **Figure 6.** *Agent location in hierarchical networks.*

In the model of mobile agents, data stay at the local place while the processing task is moved to the data locations. The management functions are executed locally, and only the resulting data are sent to the manager. By transmitting the code instead of data, the mobile agent model offers several important benefits: reduction in network bandwidth requirements, which is especially important for real-time applications and when communication uses low-bandwidth wireless channels; an agent can migrate to another node when the hosting node is compromised; network scalability is supported; an agent can migrate to regions of interest independent of the movement of nodes, if they are mobile; extensibility is supported, that is, mobile agents can be programmed to carry task-adaptive processes that extend the capability of the system; more stability, because mobile agents can be sent when the network connection is alive and return results when the connection is reestablished

The MANNA information architecture is based on the object-oriented information model. Basically, the system is decomposed into two categories of modules, which play the role of managers and agents exchanging management information.

along with the network data; it reduces delay in management actions; managers are not required to instruct agents all the time; the main management part does not reside only in the manager; and agent cloning offers robustness and fault tolerance.

INFORMATION ARCHITECTURE

The MANNA information architecture is based on the object-oriented information model. Basically, the system is decomposed into two categories of modules, which play the role of managers and agents exchanging management information.

The *information model* provides mapping of manageable resources and support of object classes. The information model contains object classes that represent e resources under different levels of abstraction, being functional areas, management levels, and network functionalities.

The design of an information model for a WSN is a complex task. The solution of the MANNA architecture to tackle this complexity is the abstraction represented in Fig. 2.

There are two types of object classes defined in the MANNA architecture: managed objects and support objects. The *managed object* class directly relates with the network components and with the network itself. On the other hand, the *support object* classes play the role of supporting management functions (i.e., making available to them the necessary information).

The specification of an object class is done through predefined syntactic structures called *templates* that utilize *Abstract Syntax Notation.1* (ASN.1)s to describe the objects and their characteristics.

The object classes may be inherited or reused from standard objects. Reuse allows future management integration. Some object classes and their new attributes, based on WSN characteristics, are listed below.

Support Object Classes — These classes can be programmed in the agent or present in the management application. These classes are mostly derived from Open Systems Interconnection (OSI). Some support object classes are log, stateChangeRecord, attibuteChangeValueRecord, alarmRecord, eventForwardingDiscriminator, and managementOperationSchedule.

Managed Object Classes — Observing the functionalities of WSNs, the following object classes can be identified.

Network: Composed of interconnected managed objects (physical or logical ones), capable of exchanging information. Examples of new attributes: network identifier, composition type (homogeneous, heterogeneous), organization type (flat, hierarchical), organization period, mobility (stationary, stationary nodes and mobile phenomenon, mobile node, and mobile phenomenon), data delivery (continuous, event driven, on demand, programmed), type of access point (sink node or base station), and localization type (relative and absolute).

Managed element: Represents the sensor and acting nodes or other WSN entities, which execute functions on managed elements, providing sensing, processing, and communication services. Examples of new attributes: localization (relative or absolute), element type (common node, sink node, gateway, cluster head), minimum energy limit, and mobility (direction, orientation, and acceleration).

Equipment: Represents the physical aspects of the sensor node constitution, which is composed of memory, processor, sensor device, battery, and transceiver. The equipment class can be specialized in object classes: battery (with the attributes of battery type, capacity, remaining energy level, energy density, max current), processor (clock, state of use, available memory, endurance, AD channel, operating voltage, IO pins), sensor (sensor type, current consumption, voltage range, minmax range, accuracy, temperature dependence, version, state current), and transceiver (type, modulation type, carrier frequency, operating voltage, current consumption, throughput, receiver sensitivity, transmitter power).

System: Represents a set of hardware and software that constitutes an autonomous system capable of executing information processing and/or transference. Examples of new attributes: operational system type, version, code length, complexity, and synchronization type (mutual exclusion, synchronization of processes). A notification of change in an attribute value must be reported upon an event occurrence, such as software upgrade.

Environment: Represents the environment where the WSN is operating. Examples of new attributes: environmental type (internal, external, and unknown), noise ratio, atmospheric pressure, temperature, radiation, electromagnetic field, humidity, and luminosity. The environment can present static and dynamic features.

Phenomenon: Represents the phenomenon behavior in the environment where the WSN is operating. Examples of attributes: phenomenon type, occurrence frequency, and media type.

Connection: Represents the actual connections and are expressed as an association between particular points. The direction of connectivity can be unidirectional (asymmetric) or bidirectional (symmetric). If an instance of this class is unidirectional, point a will be the origin and terminal point z will be the destination. The operational state will indicate the capacity to load a signal. An example of an attribute for this class is the communication type (simplex, half duplex, full duplex).

PUTTING IT ALL TOGETHER

Consider that a managing entity has just received a sensing range area map and detects the existence of high node density, because there are lots of intersections from the sensing range of the nodes. The managing entity faces a redundancy problem of the sensing data received. On one hand redundancy provides a mechanism for fault-tolerance and multi-resolution, on the other hand, it represents waste of resources.

This redundancy problem was detected by the MANNA architecture using the WSN models, in particular, the sensing coverage area map. Based on this map, maintenance functions may be executed. These functions can be manual, automatic, or semi-automatic, depending on the physical architecture established for the management and the management policy. In this case, a function possibly invoked is the node operating state control function.

This function represents the intersection of the three abstraction plans for the configuration functional area, network element management level, and sensing functionality. The function allows placing the redundant nodes in the inactive state. For this, the agent attributes the value *disable* for the operational state of the objects (present in the MIB) that represent such nodes, acting over the nodes and removing them from the sensing service.

In the MANNA architecture, the execution of management services (composed of functions) is dependent on the information obtained from the WSN models (topology map, energy map, covering area map). The definition of functions that compose these services is based on the three functional plans.

CONCLUSION

Wireless sensor networks represent a new frontier in the development of technology to be used in a variety of applications of our daily life in the future. As a new research area, there are several open problems that need to be investigated. One of them is management of those networks. As pointed out earlier, there are several significant differences in the management of traditional networks and WSNs. Therefore, we need a different management architecture for this kind of network.

The task of building and deploying management systems in environments where there will be tens of thousands of network elements with particular features and organization is very complex. The task becomes worse due to the physical restrictions of the sensor nodes, in particular energy and bandwidth restrictions.

This work presents and discusses the MANNA management architecture for WSNs, based on the principles presented and discussed earlier. The article discusses the management functional areas, WSN models, WSN functionalities, and management levels. It presents the technical basis to the evolution of such a technology from the management point of view.

As mentioned before, a WSN is application-dependent, which implies that the management requirements also change among sensor networks. Nevertheless, the MANNA architecture provides flexibility when defining the three architectures: functional, information, and physical. The coordination among the three planes is based solely on policy-based management. The functional architecture allows the establishment of all possible configurations for the management entities (manager, agent, and MIB). The information architecture specifies object classes and the syntax and semantics of the information exchanged among the entities. The physical architecture reflects the flexibility provided by the functional architecture by allowing different locations of managers and agents, and the definition of a centralized, distributed, or hierarchical architecture. It also establishes the communication interfaces for the management entities according to the available protocol profiles.

In the MANNA architecture, the execution of management services (composed of functions) is dependent on the information obtained from the WSN models. The definition of functions that compose these services is based on the three functional plans.

REFERENCES

[1] S. Lindsey *et al.*, "Data Gathering in Sensor Networks Using the Energy Delay Metric," *IEEE Trans. Parallel and Distrib. Syst.*, vol. A, no. 13, Sept. 2002, pp. 924–35.
[2] D. Estrin, R. Govindan, and J. Heidemann, "Embedding the Internet," *Commun. ACM*, vol. 43, no. 5, May 2000, Special I, pp. 39–41.
[3] B. R. Badrinath *et al.*, "Special Issue on Smart Spaces and Environments," *IEEE Pers. Commun.*, Oct 2000.
[4] S. Meguerdichian *et al.*, "Coverage Problems in Wireless AdHoc Sensor Networks," *INFOCOM*, 2001, pp. 1380–87.
[5] S. B. B. Deb and B. Nath, "A Topology Discovery Algorithm for Sensor Networks with Applications to Network Management," Tech. rep. DCSTR-441, Dept. of Comp. Sci., Rutgers Univ., May 2002.
[6] M. W. Subbarao, "Ad Hoc Networking Critical Features and Performance Metrics," Tech. rep., Wireless Commun. Technology Group, NIST, Sept. 1999.
[7] D. B Johnson and D. A. Maltz, "Dynamic Source Routing in Ad Hoc Wireless Networks," Imielinski and Korth, Eds., *Mobile Comp.*, vol. 353, 1996.
[8] W. Chen, N. Jain, and S. Singh, "Anmp: Ad Hoc Network Network Management Protocol," *IEEE JSAC*, vol. 17,no. 8, Aug. 1999, pp. 1506–31.

BIOGRAPHIES

LINNYER BEATRYS RUIZ (linnyer@dcc.ufmg.br) has been an associate professor of computer science at the Pontifical Catholic University of Paraná (PUCPR), Brazil, since 1996. She is currently working toward a Ph.D. degree in computer science at the Federal University of Minas Gerais (UFMG), Brazil. She received an M.S. degree in electrical engineering and industrial information from the Federal Center of Technological Education of Paraná (CEFETPR), Brazil, in 1996 and a B.S. degree in computer engineering from PUCPR. Her areas of interest and research include WSNs, telecommunications and management of computer networks, information theory, and software development. She is an expert in telecommunications management network (TMN). Since 1993 she has participated in and coordinated research groups on TMN. Currently, she is coordinating the WSN group in the Computer Science Department of UFMG. She has been a member of the technical program committees of IEEE Latin American Network Operation (LANOMS 2001) and a referee in other conferences.

JOSÉ MARCOS NOGUEIRA [M] (jmarcos@dcc.ufmg.br) is an associate professor of computer science at UFMG, Brazil. His areas of interest and research include computer networks, telecommunications and computer network management, and software development. He received a B.S. degree in electrical engineering, an M.S. degree in computer science from UFMG in 1979, and a Ph.D. degree in electrical engineering from the University of Campinas, Brazil in 1985. He held a post-doctoral position at the University of British Columbia, Canada, 1988–1989. He headed the Department of Computer Science at UFMG from 1998 to 2000. Currently, he heads the computer network group at UFMG [PB1] and was technical coordinator of the System for the Integration of Supervision (SIS) [PB2] Project where a complex and distributed system for the management of telecommunications networks was developed. He has served in various roles, including General Chair (1985) and TPC Chair (1999) of the Brazilian Symposium on Computer Networks (SBRC), and General Chair of LANOMS 2001. He has been a TPC member in IEEE/IFIP NOMS (2000 and 2002), IEEE/IFIP IM 2003, IEEE LANOMS (1999 and 2001), IEEE/IFIP MMNS (2000, 2001, 2002, 2003), IPOM 2002, IFIP/IEEE IM (2003), SBRC(from 1990 to 2003), and IEEE/IFIP DSOM 2003.

ANTONIO A. F. LOUREIRO (loureiro@dcc.ufmg.br) is an associate professor of computer science at UFMG, Brazil. His areas of interest and research include computer networks, network management, distributed algorithms, mobile computing, and wireless communication. He received B.S. and M.S. degrees in computer science from the UFMG, and a Ph.D. degree in computer science from the University of British Columbia, Canada, in 1995. He has served in various roles, including General Chair of the Brazilian Symposium on Computer Networks — SBRC 2000 and TPC Chair of LANOMS 2001.